HARDWARE AND SOFTWARE IMPROVEMENTS TO A
PACED DATA ACQUISITION SYSTEM
FOR TURBOMACHINES

Patrick Anthony McCarville

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

HARDWARE AND SOFTWARE IMPROVEMENTS
TO A PACED DATA ACQUISITION SYSTEM
FOR TURBOMACHINES

by

Patrick Anthony McCarville

June 1981

Thesis Advisor: R. P. Shreeve

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Hardware and Software Improvements to a Paced Data Acquisition System for Turbomachines		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; June 1981
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Patrick Anthony McCarville		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE June 1981
		13. NUMBER OF PAGES 69
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Programmable; synchronized sampling; digital phase lock loop, direct memory access

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Modification of the phase lock loop synchronizing circuits and of the method of input/output communication used in a synchronized data sampling system, are reported. A device known as PACER which used an analog phase lock loop for synchronization and produced a non linear set of synchronizing pulses, was modified to use a CMOS digital phase lock loop, resulting in a linear set of pulses. The associated programming which controlled the data

acquisition process and sequencing, was changed to use the direct memory access feature of the system computer. This enabled data, from high response pressure transducers mounted in a turbomachine, to be taken once every rotor revolution rather than once every ten revolutions. A user's manual for paced data acquisition is included.

Hardware and Software Improvements to a Paced Data Acquisition System for Turbomachines

by

Patrick Anthony McCarville
Lieutenant Commander, United States Navy
B.S., University of New Mexico, 1972

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1981

ABSTRACT

Modification of the phase lock loop synchronizing circuits and of the method of input/output communication used in a synchronized data sampling system, are reported. A device known as PACER which used an analog phase lock loop for synchronization and produced a non linear set of synchronizing pulses, was modified to use a CMOS digital phase lock loop, resulting in a linear set of pulses. The associated programming which controlled the data acquisition process and sequencing, was changed to use the direct memory access feature of the system computer. This enabled data, from high response pressure transducers mounted in a turbomachine, to be taken once every rotor revolution rather than once every ten revolutions. A user's manual for paced data acquisition is included.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

SYMBOLS

A	Driver Amplifier
B	4-Bit binary counter
C	Comparator
F	Buffer Amplifier
U	AND gate
L	Latching Flip Flop
D	Delay Flip Flop
I	Inverter
17_8	PACER I/O controller port
11_8	A/D I/O controller port

ABBREVIATIONS

A/D	Analog-to-Digital
I/O	Input-Output
RTE	Real-Time Executive
1/Rev	Once per Revolution
1/BL	Once per Blade Passage
PLL	Phase Lock Loop
CMOS	Complementary Metal Oxide Semiconductor
TTL	Transistor-Transistor Logic
DMA	Direct Memory Access
DCPC	Dual Channel Port Controller
TP	Test Point

ACKNOWLEDGEMENT

To my thesis advisor, Dr. Raymond P. Shreeve, goes my sincere appreciation for his steadfast moral and technical support. Without his timely and intuitive guidance, this project would not have culminated in the results reported herein.

Secondly, to Mr. Jack King goes credit for the fine workmanship resulting in the circuits built and tested for this report. His expertise in the field of electronics was an invaluable aid in the completion of this work.

Finally, to Mr. Alan McGuire goes my gratitude for his most professional work in the drawings and figures completed for this report.

I. INTRODUCTION

The device described herein and referred to as the "PACER" is part of a computer controlled data acquisition system in use at the Turbopropulsion Laboratory at the U.S. Naval Postgraduate School. It is an electronic interface unit built of solid state and integrated circuit components. The PACER was designed to allow the acquisition of data from high response transducers mounted in the case of rotating machines to be synchronized with respect to rotor position. Using the PACER, the analog to digital conversion of the data from a particular transducer can be programmed to occur at any position of the rotor with respect to the transducer, independent of rotor speed.

The PACER was first designed and built in a "bread board" configuration in 1976 by James C. West as described in Reference (1). U.S. patent no. 4,181,962 was issued for the PACER on January 1, 1980. The present hardware configuration of PACER involves minor but important changes which improve its performance and are documented in this report.

The original PACER made it difficult for the typical user to acquire accurate data in a reasonable amount of time for the following three reasons:

(1) The timing pulses generated within PACER were not always spaced linearly in time between blade pair

synchronizing pulses. This resulted in data which in some cases was subtly distorted, and in other cases appeared to have noise riding on it.

(2) The range over which the PACER could follow rotor RPM changes and remain synchronized was limited to approximately $\pm$ 15% of the initial RPM at which the PACER was set to take the data. This required repeated, and somewhat involved, manual adjustment of an RPM "lock-on" procedure to acquire data at different speeds.

(3) The rate at which data could be taken was limited below the desired rate. This meant that rather than being able to sample data on every revolution of the rotor, the system was only capable of taking data once every 8 to 10 revolutions, depending on RPM.

The methods used to improve the performance of the PACER fall into two areas, hardware and software. Hardware changes were used to improve PACER linearity, eliminate manual lock-on procedures, and increase speed-following range. A change in acquisition software was used to increase the rate at which data could be taken.

The change in PACER hardware consisted of replacing the original analog 562 phase lock loop with a CMOS digital phase lock loop and eliminating the discrete components forming the coupling circuit in the PLL feedback path. The change to acquisition software involved use of the DMA (direct memory

access) feature of RTE-IVB system software [Ref. 2] which is incorporated in the I/O driver written for the PACER.

As a result of the hardware and software changes which were made, all of the limitations described above were eliminated. The improvement in PACER performance was verified using test programs and rotating machine signal simulation circuits which enabled controlled test techniques to be employed.

In the following section of this report a description of the entire paced data acquisition system is given. Section III describes the changes made to PACER hardware and the effects of those changes, while Section IV describes the change to acquisition software. In Section V the results of the changes are verified with a report of the system tests. Section VI lists conclusions and recommendations for further system development. Appendix A contains detailed hardware circuit design figures and Appendix B details the software programs - both acquisition FORTRAN and system assembly language drivers. Finally Appendix C is a step by step system users manual for paced data acquisition.

II. PACED DATA ACQUISITION SYSTEM DESCRIPTION

A. GENERAL DESCRIPTION

Components of the system are shown schematically in Figure 1 and details of the circuits, including the modifications made in the present work are shown in Figures 2-4. The PACER acts as a secondary controller on the interface between the Hewlett-Packard HP 21-MX computer and the A/D converter. Referring to Fig. 1, in a normal (not paced) data taking sequence, the 21MX would call on the A/D converter to take an analog data sample, convert it to digital, and output it to the computer memory. Since the computer program execution cannot be synchronized to the rotation of the machine shaft, the data sample would be from a random, unidentified point.

In a paced data acquisition sequence, the PACER provides the timing control to the 21MX computer. After the 21 MX computer passes a word (IBLADE) to the PACER defining the desired position, the PACER acts as an intermediary. It intercepts the computer command to the A/D converter, tells the computer that the A/D converter is in the process of acquiring the data, then sends a command pulse to the A/D converter at a time synchronized to the desired position in the cycle of the rotation of the machine.

The sequence of events for paced data acquisition using the software developed in the present work is as follows:

(1) The user enters the main program (which was written to be used for system testing or for data acquisition).

(2) The main program prompts the user for information regarding the (rotor) position(s) desired at which to start taking data points. This information defines the integer IBLADE.

(3) The main program calls the PACER, passes IBLADE to the PACER, and receives rotation speed (IRPM) from the PACER. Control then returns to the main program.

(4) The main program calls the A/D converter telling it to take a number of data samples (N) (at the desired point). When complete, control is returned to the main program.

(5) If a survey of positions (for example, across a pair of blade passages) has been programmed using a DO loop, the main program repeats steps (2) and (3), incrementing IBLADE each time until the loop is finished.

(6) When all data have been taken and stored in the computer memory, the main program converts the digital data (which are binary whole numbers) to decimal values scaled appropriately to the ± 1.0 volt range of the A/D converter. As programmed, it then outputs that data to the desired peripheral(s) (i.e. the printer, plotter, or terminal).

B. SYSTEM SOFTWARE

The software used in the data acquisition should be viewed as consisting of two separate parts, the RTE-IVB operating system which is generated in-house following standard procedures supplied by Hewlett-Packard, and the system test and operation FORTRAN program which may be modified at any time by the user operating in the RTE-IVB system.

1. RTE-IVB Operating System

The RTE-IVB (Software) Operating System is generated (and can be regenerated) by the System Manager in a process which "configures" the System for the particular set of (I/O) devices which the computer must address [Ref. 3]. RTE-IVB permanently resides on disc and is automatically loaded when the system is turned on. It consists of a collection of software modules which perform system resource management, operator requests for utility programs (FORTRAN compiler, file editor, etc.), and user program scheduling for time sharing [Ref. 4]. RTE-IVB is visible to the user through interaction at the terminal. It allows multi-programming through its scheduling modules so that more than one user's program may be active at a time. The input/output (I/O) drivers are a set of modules in the RTE-IVB System. They are the software routines which control the input and output communication between the user's program and addressed peripheral devices. The drivers enable efficient use of peripherals which act at different speeds by allowing one or more fast I/O requests to be

processed while waiting for a request from a slow peripheral device to be completed. A driver written for the PACER (DVR.70) and a driver written for the A/D converter (DVR.56) are part of RTE-IVB and are listed in Appendix B.

2. System Test and Operation Program

The system test and operation program (A2D) is a FORTRAN program written and used in the course of the present work. A listing and flow diagram are given in Appendix B. Program A2D converts the user's requests, which are entered at the terminal, to the parameters required by the RTE-IVB I/O drivers. It is an interactive program consisting of two parts. The first part, a system test (subroutine ADTES), is entered if the user wants to carry out a test of the paced data acquisition system simply to ensure that all components are operating correctly. The second part, (Subroutine RPACE), is executed if paced data is to be acquired from a test rig. Both the "test" and "operation" portions of A2D use the FORTRAN statement "CALL EXEC" to enter the appropriate driver. The CALL EXEC statement, with its accompanying parameters, transfers control from the FORTRAN program to the assembly language driver for the device requested. A simplified flow diagram of the CALL EXEC routine is shown in Figure 5. The driver initiates the input or output task as specified in the parameters which it received. If the task is for "output", after the task is initiated control may return to the calling FORTRAN program or another user's program. If the task

requires "input", then control may be passed to another program, but not back to the calling program, since the calling program must have an input value to continue executing. This permits efficient use of the computer's time, which is essential for multi-programming, while waiting for a slow peripheral device to complete its cycle of operation.

C. SYSTEM HARDWARE

The hardware devices used in paced data acquisition are the HP-21MX computer with printer, its magnetic disc, plotter, and terminal, the HP 5610A A/D converter and the PACER.

1. Hewlett Packard HP 21 MX Computer

The HP 21 MX is a (Micro-programmable) mini-computer having 128 machine instructions and 32K of logical main frame memory. In the present configuration a 20 megabyte capacity disc and disc operating system are an integral part of the system. A detailed description of the computer is given in Reference 2.

An important feature which is typical of computers of this size is the input-output structure. With a limited number of relatively slow I/O devices to be serviced, the computer can communicate with all devices through a single port known as the I/O bus. Each device requires its own I/O interface on the bus. The interface acts as a filter and ensures that output information is received only by the device designated to receive it and that input information is put

on the bus from only one device at a time. The I/O software drivers control the I/O hardware interfaces by commands to either "turn-on" or "turn-off".

2. Hewlett Packard HP 5610A A/D Converter

The HP 5610A analog-to-digital converter accepts analog data input on up to sixteen different channels and under computer controlled multiplexing converts to a 10 bit binary data output. With an input conversion aperture of 50 nanoseconds, rapidly changing signals (100 KHz) can be converted accurately. The HP 5610A can operate in one of six modes as described in Reference 5. Currently the paced data acquisition system uses the "random access mode" in which a specific channel is sampled on receipt of a command word and an encode command pulse from the 21 MX computer. The command word tells the A/D converter which mode of operation to use and which channel number to sample. The encode command pulse triggers the data conversion to start 2 μsec later. The data conversion itself is finished in a total time of 10 μsec. Using computer-issued encodes, which is the mode required for paced data, the sample cycle time is 20 μsec. Hence data can be converted at rates of up to 50,000 samples per seconds, depending on how rapidly each successive command word is received.

The other mode which is used only for non-paced data is the Free Run, Random Access mode. In this mode the command word is required as before, but no encode command is

needed from the computer. The A/D converter simply converts data as fast as it can (100,000 samples per second) on the selected channel. This mode is not addressed further in this report.

3. PACER

A schematic of the PACER is shown in Figure 2. In its original form, a detailed description of the internal operation is given in Ref. 1. The PACER consists of two major sections, an "RPM counting section" and a "synchronized command pulse section". The "RPM counting section" continuously counts the number of 250 KHz time base pulses that occur between the once-per-revolution pulses received from the test rig. This number of counts is available as an output (IRPM) from the PACER on every revolution cycle.

The "synchronized command pulse" section is the heart of the PACER. It uses a phase lock loop to generate 256 pulses within each pair of blade passages (i.e. 128 pulses from blade #1 to blade #2 and 128 pulses from blade #2 to blade #3). At the same time, these pulses are counted and compared with the programmed data conversion location specified in IBLADE. When the comparison is true, a command to the A/D converter (A/D Device Command) is generated. Thus a command to convert a data sample is synchronized with a desired position of the rotation rotor in the machine.

III. CHANGES TO PACER HARDWARE

In order to determine the cause of the non-linearity in the PACER, a test chassis was built to provide easy access to the four circuit boards and to allow modifications to be attempted without interference to the working unit. The test chassis is shown in Figure 6. It is electrically identical to the system PACER shown in Figure 7 and uses the same four circuit boards. Using the test PACER with an oscilloscope it was possible to examine the wave forms, at any point in the PACER circuit. In so doing, it was found that even with the lock-on procedure recommended in Reference 1, the output pulses from the PLL (256·Fo/2) were not always linearly spaced between the beginning and end of the input pulses (Fo/2). This non-linearity is seen in the oscilloscope traces shown in Figure 8, which shows the signal at counter B1. At counter B1 the pulse frequency is 1/32 of the output frequency of the PLL which allows the non-linearity to be obvious to the eye. It was further noted that a deviation of as little as 3° from the ideal 270° phase relation called for in Reference 1, caused non-linear spacing and excessive unsteadiness ('jitter') of the pulses into counter B1. These problems were inherent in the 562N PLL when used with digital waveforms because an analog phase comparator was used in that particular circuit [Ref. 6].

A CD4046 (CMOS) PLL was therefore chosen to replace the 562N. The CD4046 uses a digital phase comparator to maintain lock [Ref. 7] and is specifically designed to operate with digital waveform inputs as are found in the PACER application. It also permits, with proper associated component design, operation over an extremely wide frequency range (by so-called frequency tracking) without loosing lock.

The changes which were made in the PLL and associated circuitry are shown in Figure 3. Both the PLL and the discrete component coupling circuits were changed. The replacement of the old coupling circuits with CMOS-to-TTL (4050B Buffer) and TTL-to-CMOS (7417 Drivers with pull-up resistors) matching devices was necessary because of the special requirements of the CMOS PLL with regard to interfacing [Ref. 7]. The detailed circuitry of the CD4046 (CMOS) PLL is shown in Figure 4. Specific details of the components are given in Appendix A.

IV. CHANGE TO ACQUISITION SOFTWARE

A. METHODS OF INPUT/OUTPUT

The two methods available under RTE-IVB for input and output are the "standard" method and Direct Memory Access (DMA). In both methods the software driver controls the initiation and completion of the I/O request. Figure 9 is a schematic representation of the hardware and software involved in an I/O request in the paced data acquisiton process. The standard I/O method requires that the software driver be entered for each data sample taken. In contrast, the DMA I/O method uses the "dual channel port controller" option of the 21 MX computer to bypass the requirement to return to the driver for each new data sample [Ref. 2]. Thus by using DMA, the time involved in executing the software driver for each sample is saved.

B. INCORPORATION OF DMA

The system software was changed so that DMA was used for the A/D I/O process. The DCPC option was added to the system in 1977. The driver DVR56 was subsequently modified by Hewlett Packard to permit DMA for I/O operation with the A/D converter. The use of the DMA feature required only that the proper parameters be specified in the CALL EXEC statement for the A/D converter. Table I lists the parameters,

with their meanings, for the CALL EXEC statements used to call the A/D converter and the PACER through the drivers DVR56 and DVR70 respectively. The parameter "N", which is passed in the call to driver DVR56, sets up the DMA option in the 21 MX I/O interface logic through the Dual Channel Port Controller (DCPC). The program A2D was written so as to use the DMA feature. A flow chart, listings, and parameters used in program A2D and the drivers DVR56 and DVR70 are given in Appendix B.

V. RESULTS

Tests were run to verify the linearity of the new CMOS PLL circuitry, to demonstrate the automatic lock on feature, and to determine the speed at which data was acquired. The tests were run using the test pulse generation circuit on circuit board #4 of the PACER. This circuit provides an electronically produced simulation of the 1/Rev and 1/Blade pulses that would ordinarily be received from the test rig. The test set up for the tests is shown in Figure 10. An external signal generator was used to provide the driving signal to the pulse generating circuit at the desired blade passing frequency. Appendix C gives detailed procedures for performing a simulation test run.

A. LINEARITY TEST

Figure 8 shows a comparison of PLL output pulses from the 562N PLL and the new CMOS digital PLL circuits. It can be seen that the new circuitry produces symmetric and evenly spaced pulses while the old PLL circuit does not.

A linear ramp test signal was input to the A/D converter on analog channel 0. The PACER test portion of program A2D was run calling for a survey across the simulated blade pair. The test was repeated for the old and new PLL circuits. Figures 11 and 12 show the output results from the PACER

using the old and new PLL circuits respectively. The apparent "bending" of the ramp test signal when seen as the graphed output from the old PLL method is due to the inherent non-linearity of the 562N PLL. The strict linearity of the CMOS digital PLL circuit was noted.

B. AUTO LOCK-ON TEST

The new CMOS digital PLL requires no lock-on procedures as did the 562N PLL [Ref. 1]. Tests were run to confirm that while varying the blade passing frequency, the new PLL remained in a locked-on condition. It was shown that within the design range of the PLL circuitry, any variation of blade passing frequency (RPM) was followed without error by the digital phase lock loop. Two separate PLL circuits were designed, each one covering a range of blade passing frequencies. One PLL circuit now covers the range from 250 Hz to 2.5 KHz. The other covers the higher range from 3 KHz to 11.1 KHz. The reasons for this division are explained in Appendix A.

C. TEST OF ACQUISITION TIME

Using the software methods used in Reference 1, a short test program calling for a specified number of data samples to be taken, was run. Clock time accurate to .1 millisec was recorded by the program just before the first sample and just after the last sample of data was acquired. The lapsed time for the total acquisition was output. It was shown that up to 10 revolutions of the machine rotor where required for each data sample to be taken.

After changing to the DMA software method described in section IV, similar tests were run. The results of these tests are shown in Table II. It was noted that the interval between samples was reduced to less than one revolution of the machine rotor.

VI. CONCLUSIONS AND RECOMMENDATIONS

The desired improvements in the paced data acquisition system were achieved; namely,

(1) The speed of acquisition of successive data samples was increased to enable data to be sampled on every revolution.

(2) The correlation between the position recorded for a paced data sample and the physical position of the probe with respect to the rotor at acquisition, was significantly improved through an improvement in the linearity and stability of the PLL and associated circuitry.

(3) The manual adjustments previously required for each small range of RPM were entirely eliminated by the reported hardware modifications.

With the present hardware and software the PACER operates as fast as is possible given the constraint that the 21 MX computer operates always in the interrupt mode for all I/O operations. If the need arises to survey across a blade pair on one resolution and the computer can be dedicated to the single task of acquiring paced data, then the non-interrupt mode of 21 MX I/O processing could be used. This change would eliminate other users during the paced data program operation. It would require that the drivers DVR56 and DVR70

to be rewritten in assembly language and loaded into the RTE-IVB operating system by the system manager. It is noted however that the maximum data rate of 100,000 samples per sec cannot be exceeded using the present A/D converter.

Table I. CALL EXEC Parameters

To call the PACER (DVR70)

CALL EXEC (1, LU, IRPM, LEN, IBLADE)

Parameter	Meaning	Limits/Value
1	I/O	1
LU	device reference number	19
IRPM	RPM timing counts returned	N/A
LEN	number IRPM of words passed	0,1
IBLADE	data position indicator	0-35,584

To clear the PACER

CALL EXEC (3, LU)

Parameter	Meaning	Limits/Value
3	clear the device	3
LU	as above	19

To call the A/D (DVR56)

CALL EXEC (1, IDRT, IBUF, N, ICHAN, ICODE)

Parameter	Meaning	Limits/Value
1	I/O	1
IDRT	device reference number	20
IBUFF	data storage array name	dimension 256
N	number of samples	1-99
ICHAN	input channel number	0-15
ICODE	mode of A/D operation	0-7

Table II. Data Acquisition Times

Run	Number Samples	RPM	Time	Time/Rev	Time/Sample
Before DMA					
1	100	17,300	1.61	.0035 sec.	.0161 sec.
2	100	17,400	1.60	.0032 "	.016 "
3	20	7,500	.51	.008 "	.025 "
4	20	8,000	.45	.0075 "	.0225 "
5	500	30,000	9.51	.002 "	.019 "
6	500	29,900	9.50	.002 "	.019 "
After DMA					
1	100	15,100	.398	.00397	.00398
2	100	15,000	.400	.004	.004
3	100	8,000	.750	.0075	.0075
4	100	30,000	.200	.002	.002

Table III. Components Used in PACER

COMPONENT	SCHEMATIC NUMBER	VALUE OR TYPE NO. High Board	VALUE OR TYPE NO. Low Board
	R1	10 KΩ	4.7 KΩ
Resistors	R2	100 KΩ	100 KΩ
	R3	1 MΩ	1 MΩ
	R4	39 KΩ	47 KΩ
	R5	12 KΩ	12 KΩ
	R6	12 KΩ	12 KΩ
	R7	10 KΩ	12 KΩ
Capacitors	C1	50 pf	.001 μf
	C2	1.5 μf	1.5 μf
Counter	B1 thru B10	74193	
Latch	L1 thru L8	7475	
Comparator	C1 thru C4	9324	
AND Gate	U1 thru U3	7408	
Inverter	I1, I2	7404	
Buffer	F1 thru F5	N4050B	
Driver	A1 thru A4	7417N	
Phase Lock Loop	PLL	CD4046	

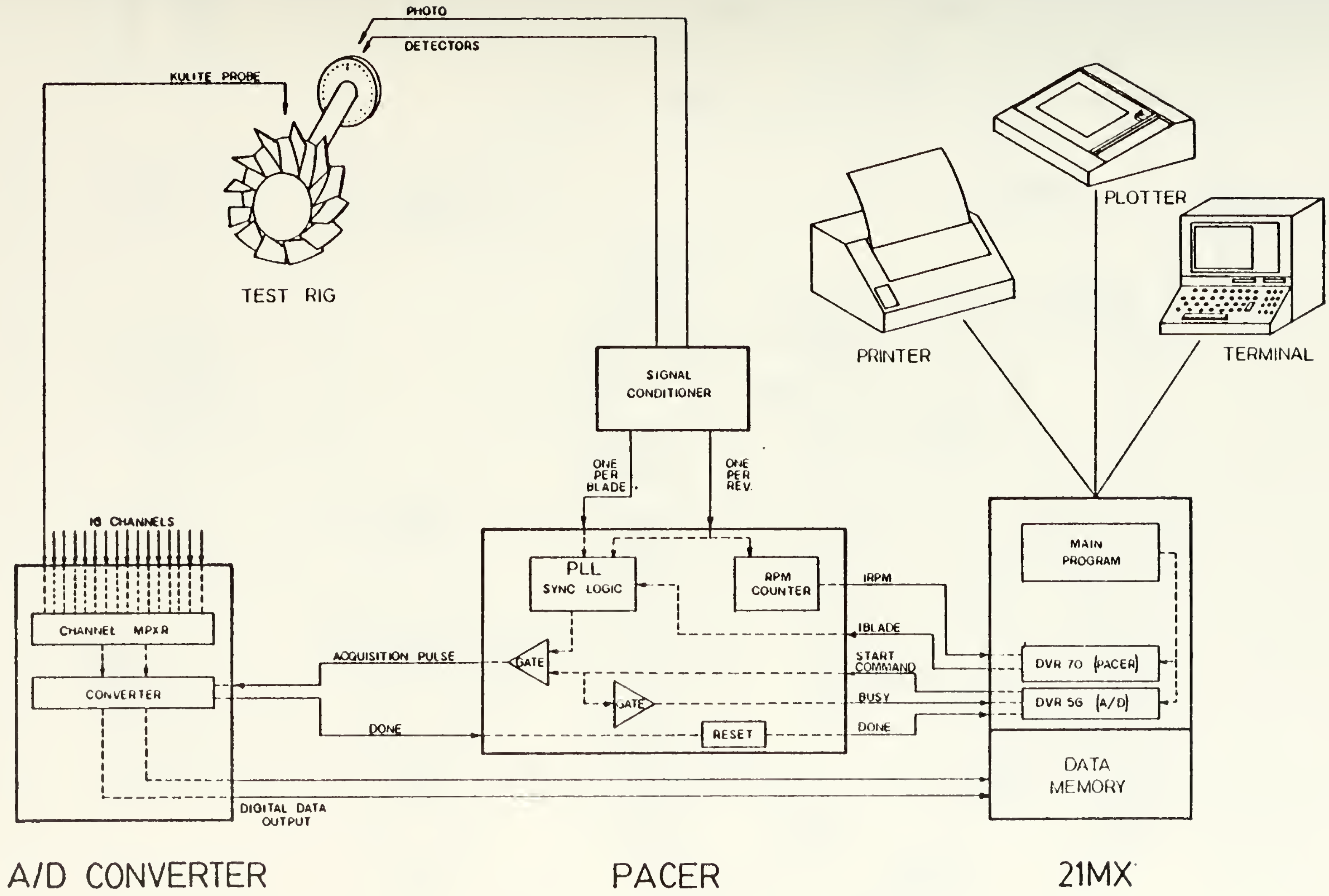


Figure 1. Paced Data Acquisition System Components

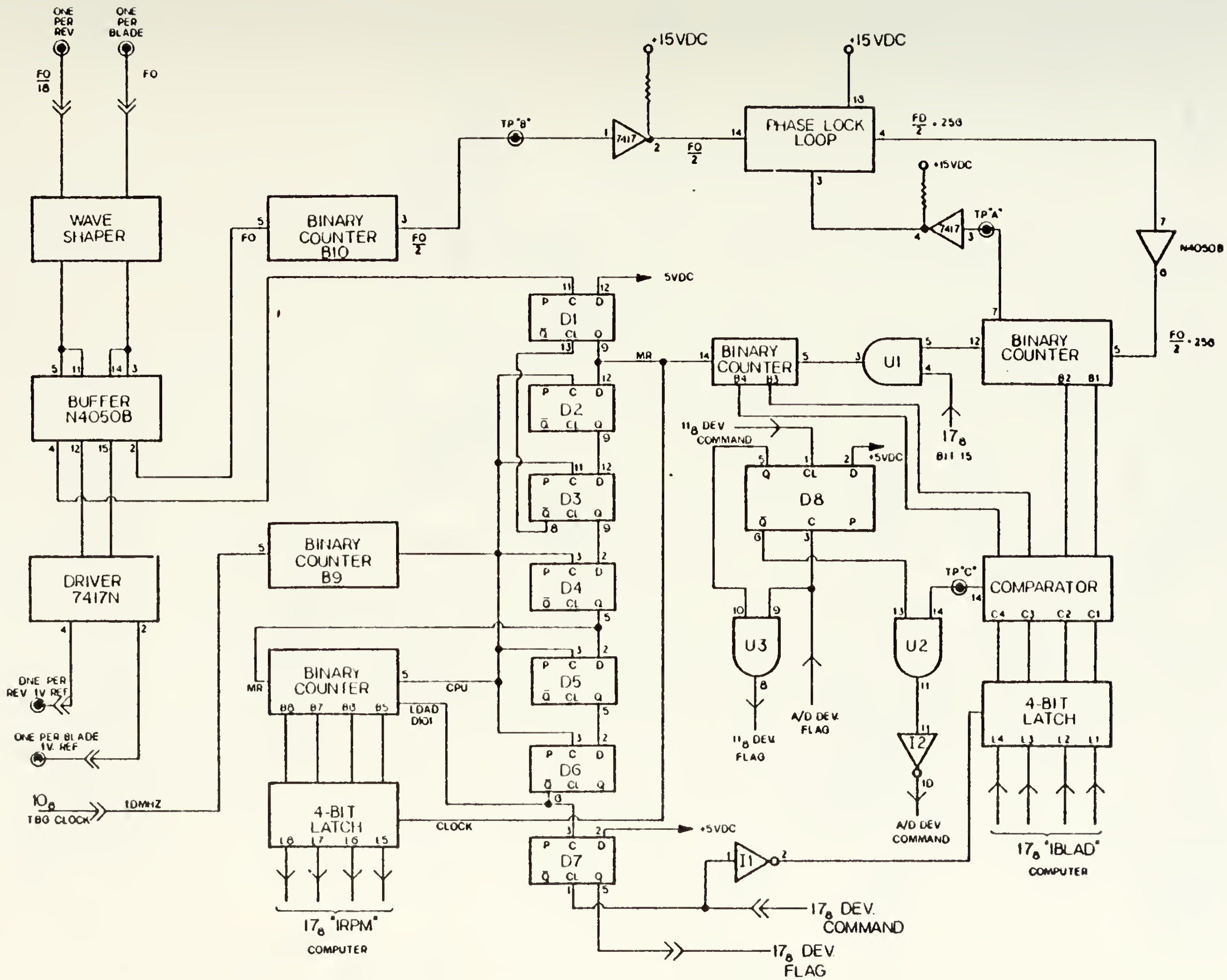


Figure 2. Schematic of PACER

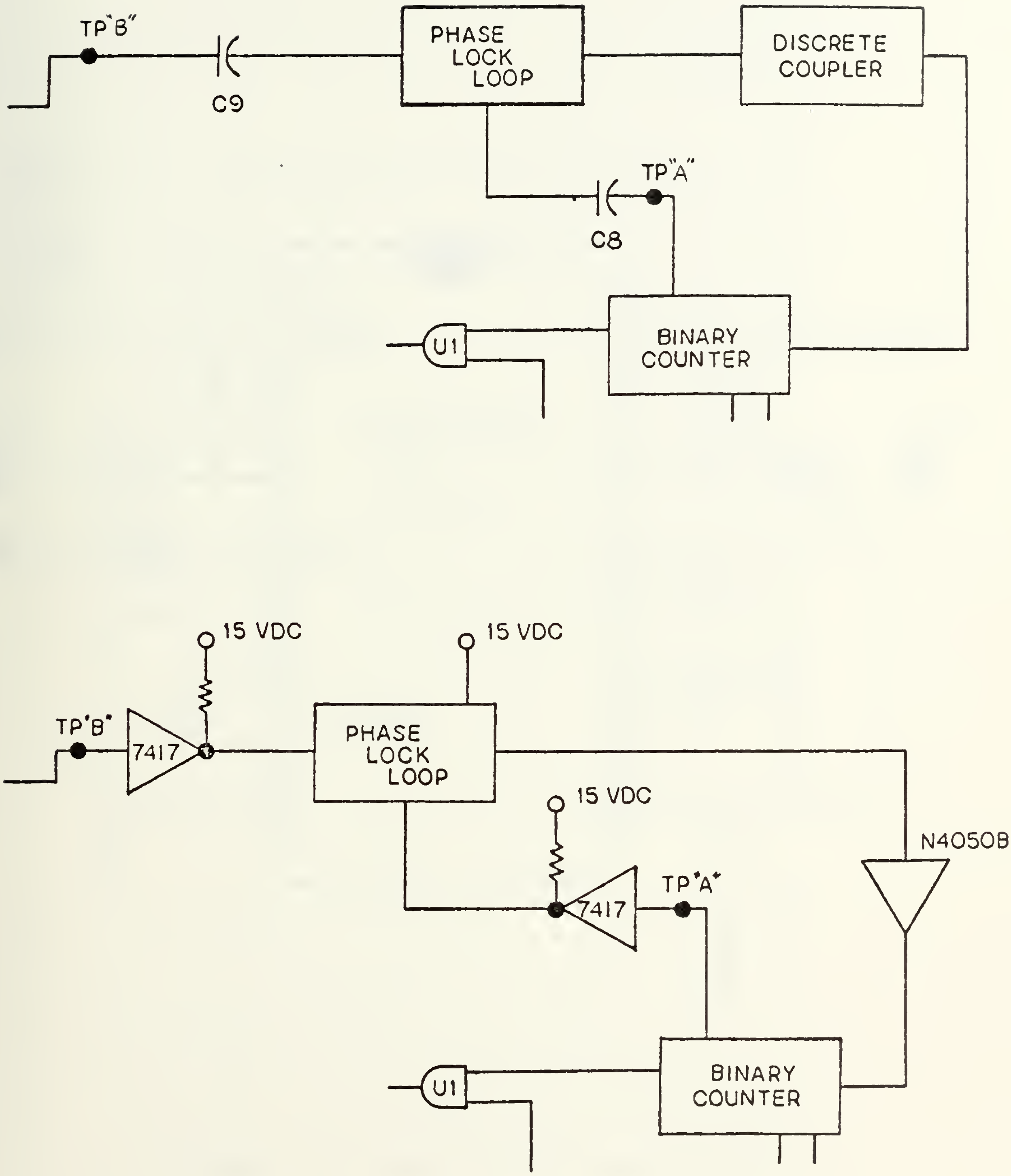


Figure 3. Original and Revised PACER Circuits

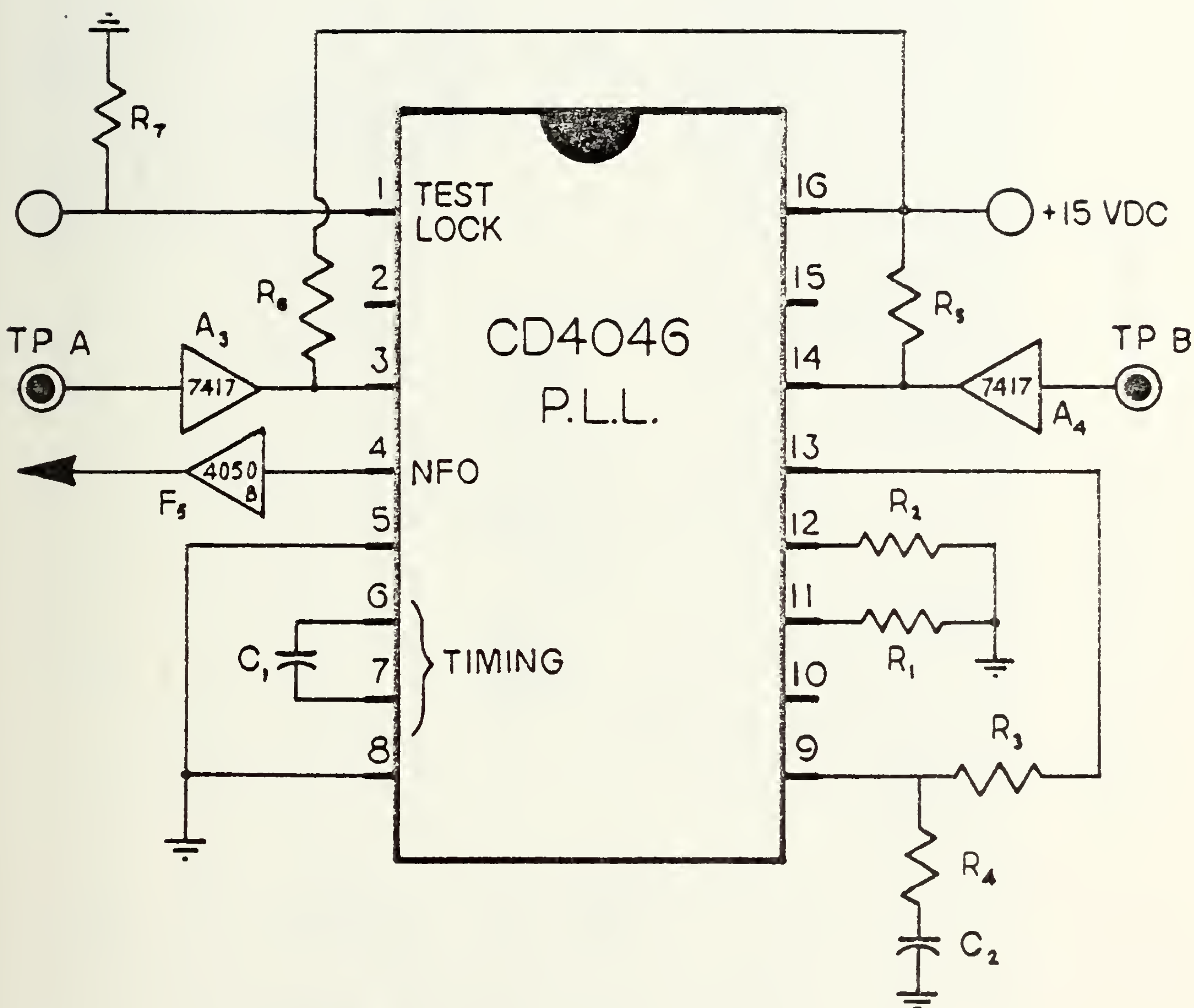


Figure 4. CD4046 PLL Circuit Detail

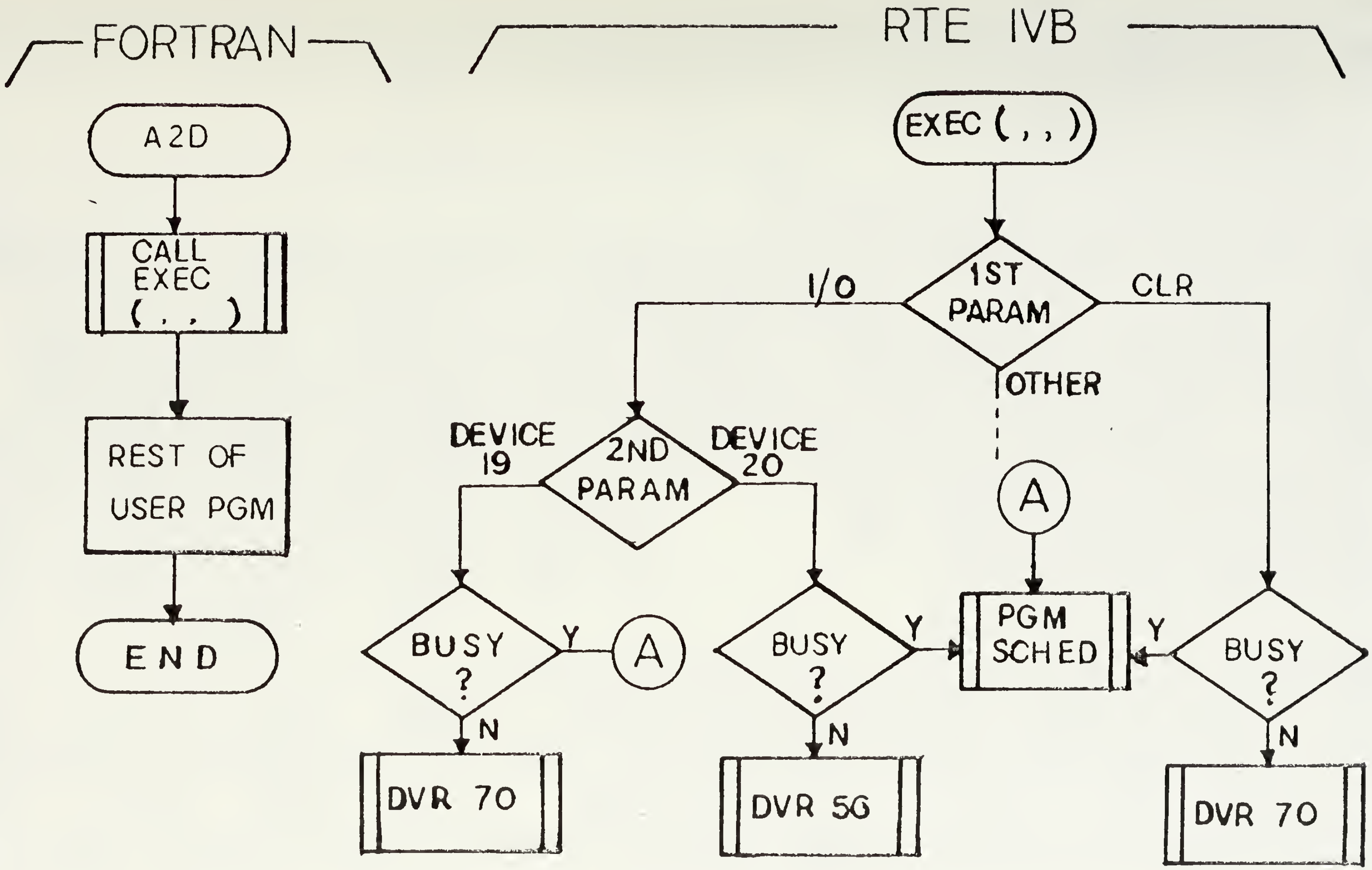


Figure 5. EXEC CALL Flow Diagram

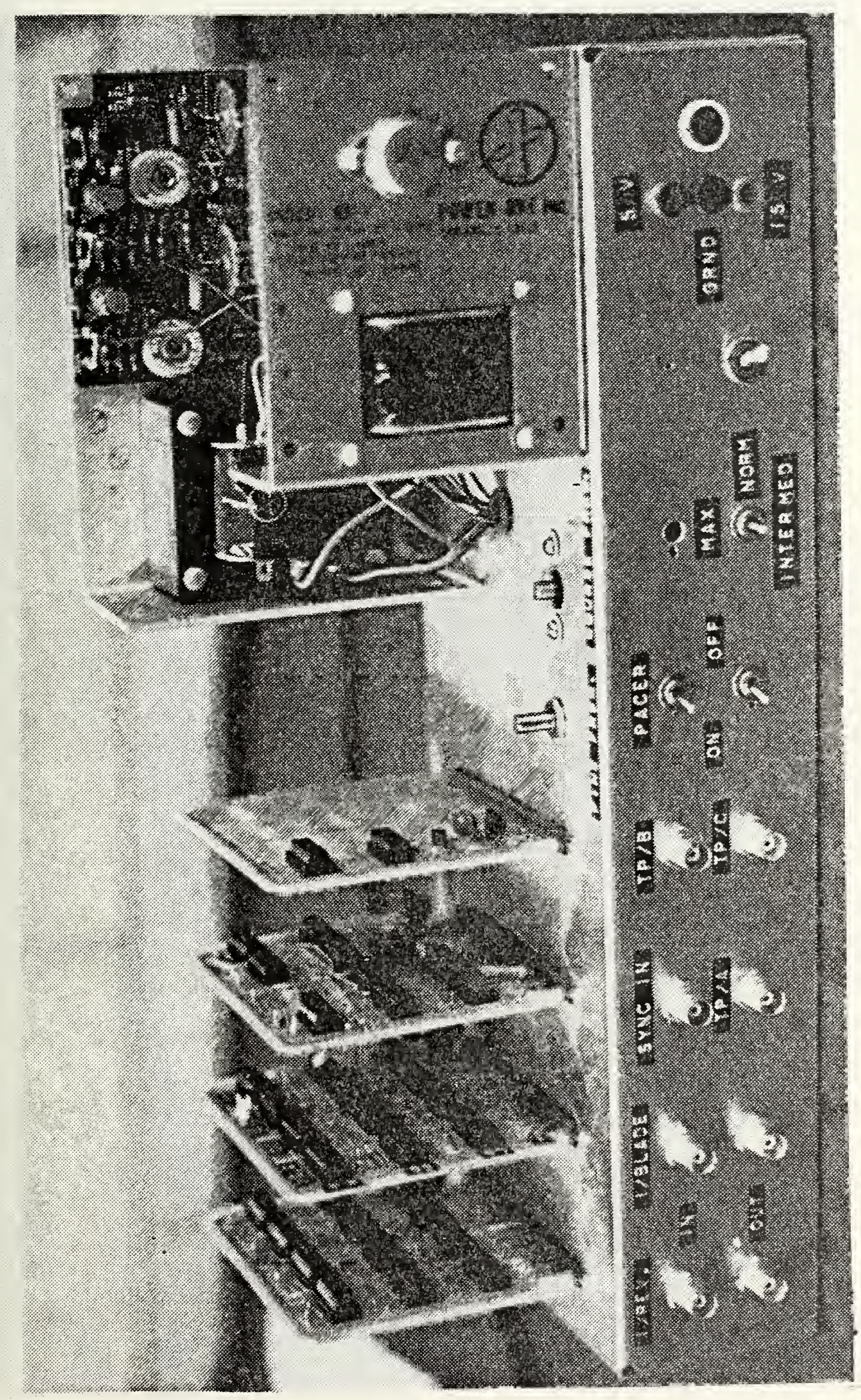


Figure 6. PACER Test Chassis

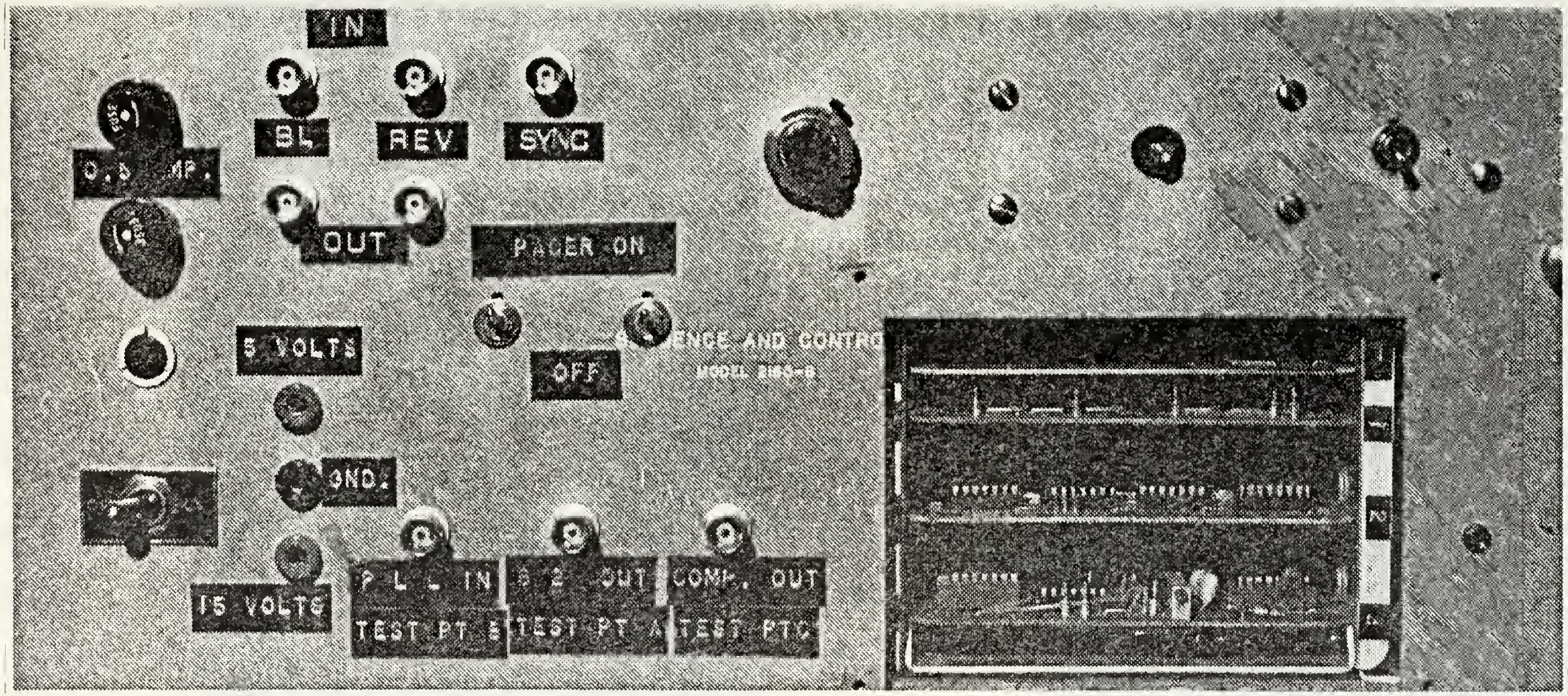
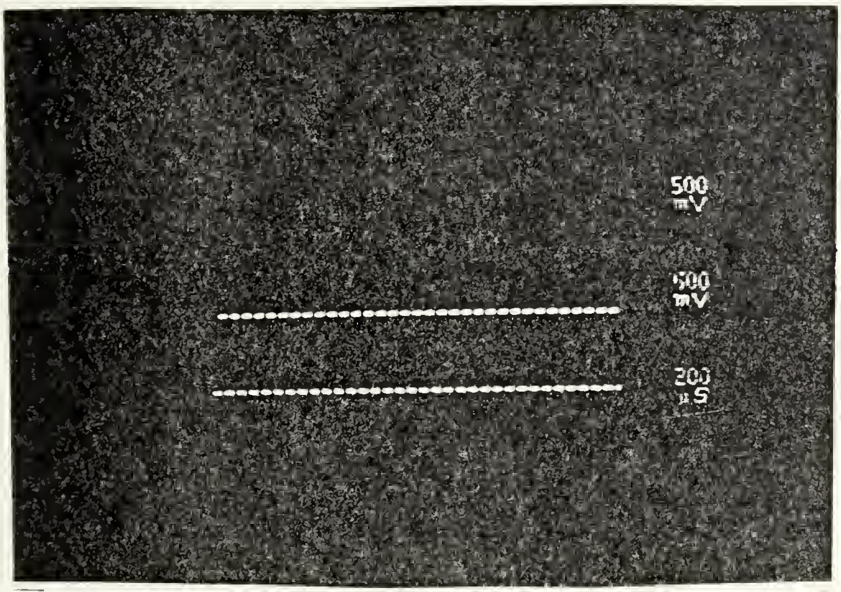
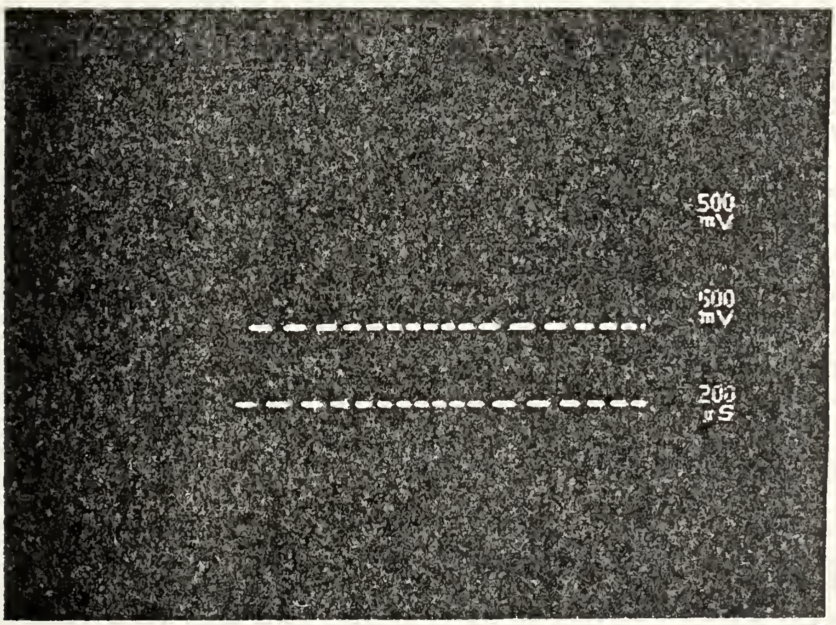


Figure 7. PACER Front Panel



(a) Digital PLL



(b) Analog PLL

Figure 8. Pulse Trains at Counter B1 for Analog & Digital PLL Circuits

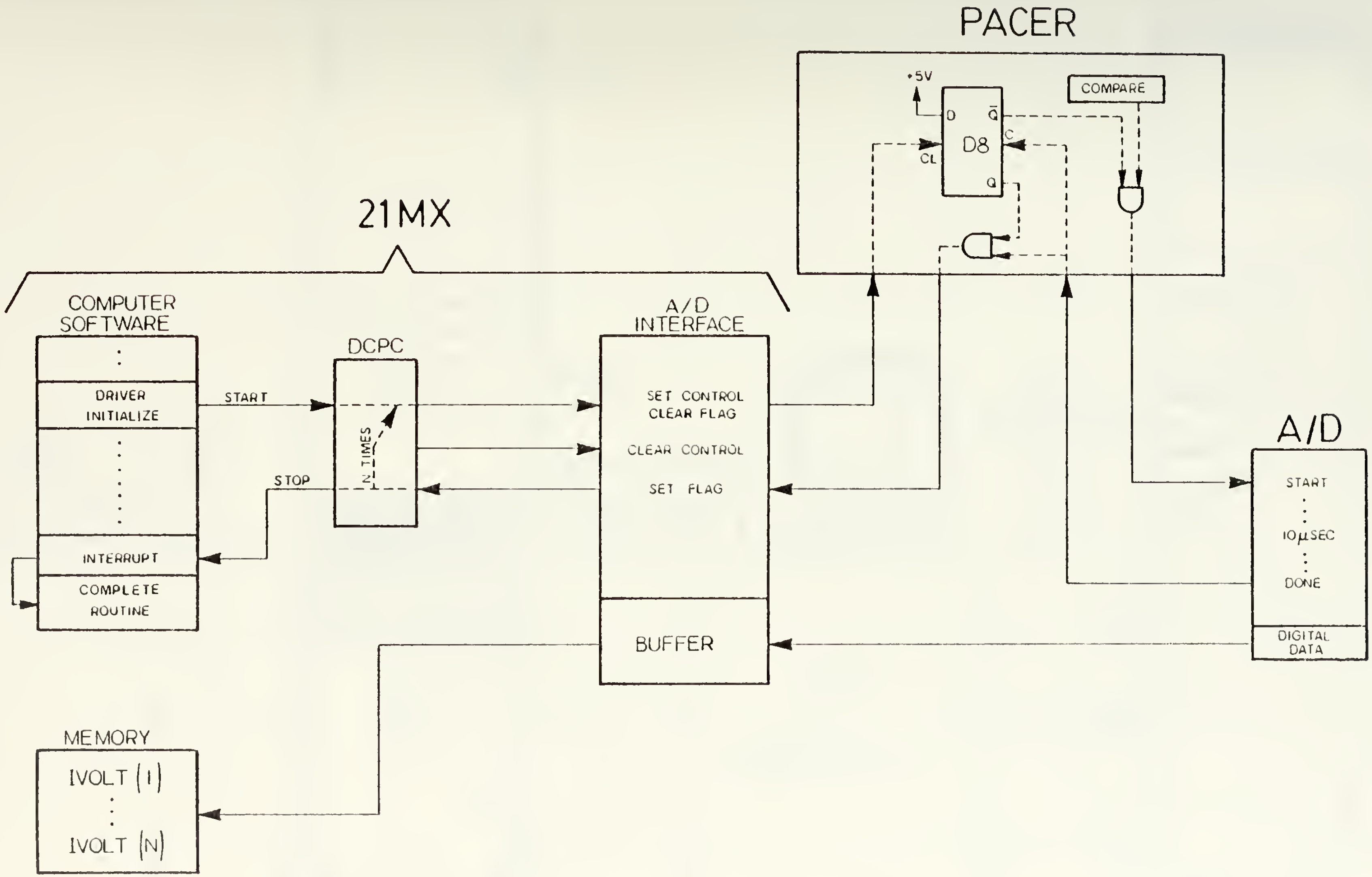


Figure 9. Paced I/O Request Flow Diagram

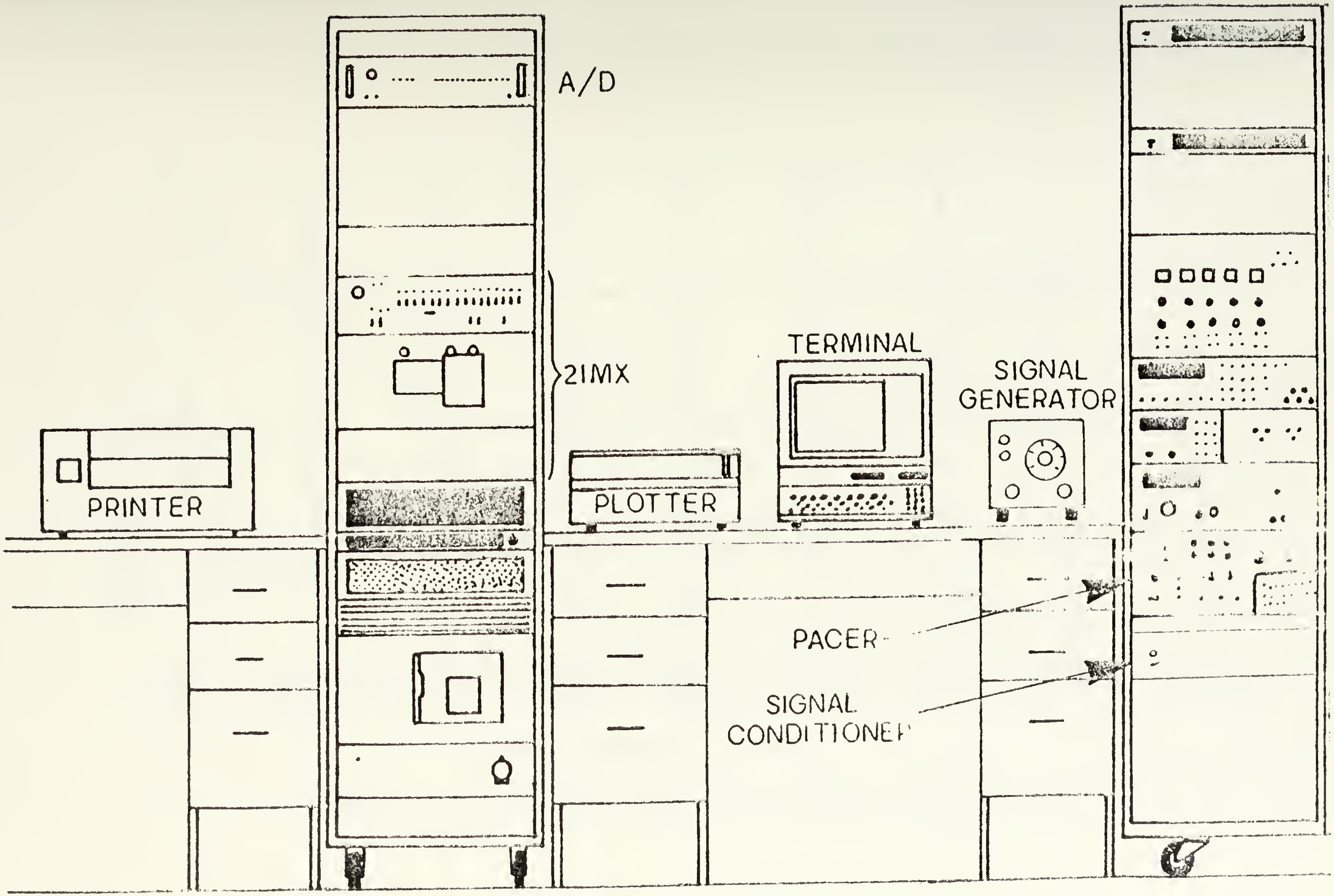


Figure 10. Data Acquisition and Test Equipment

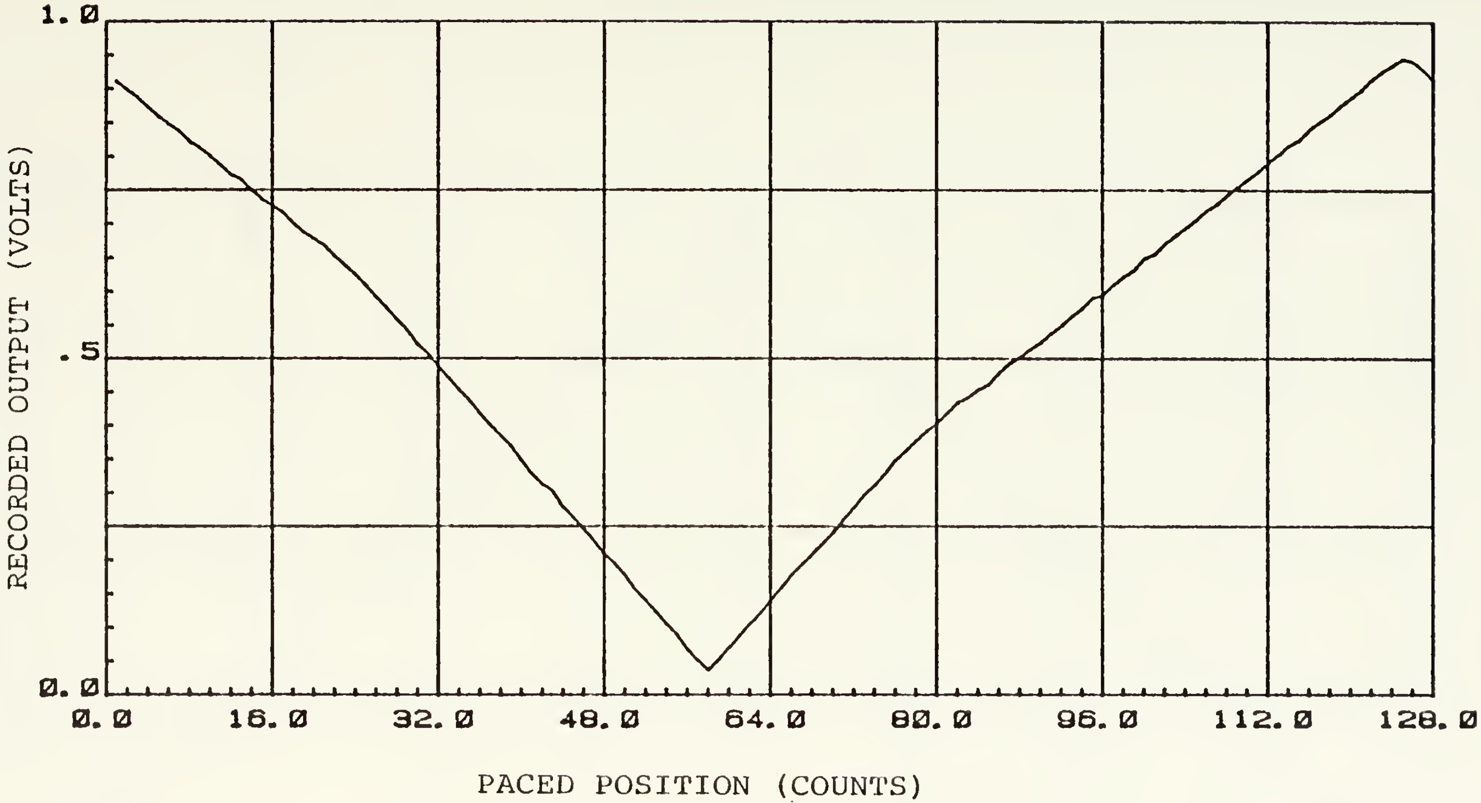


Figure 11. Ramp Test Data from Original PACER

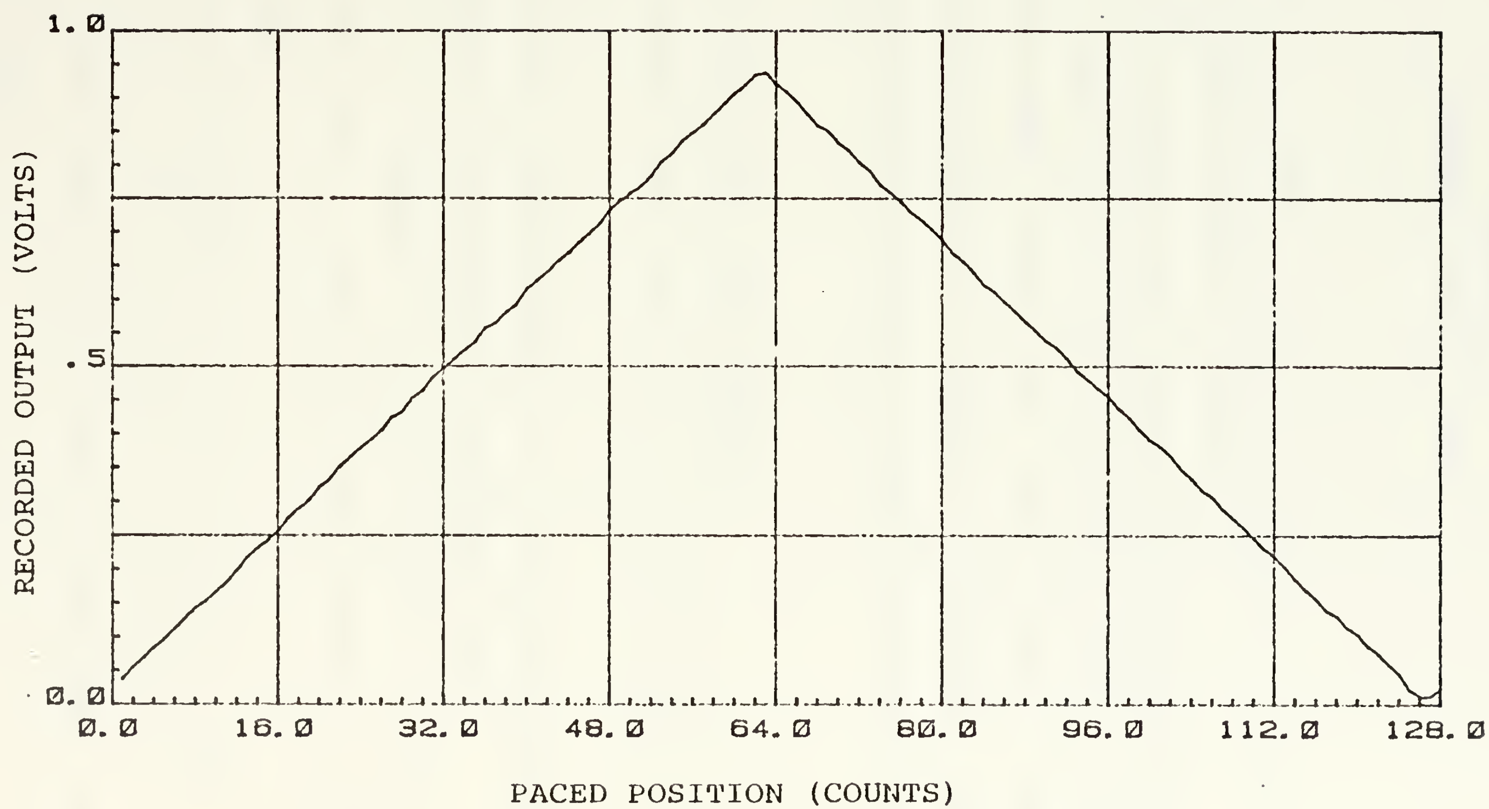


Figure 12. Ramp Test Data from Revised PACER

APPENDIX A

HARDWARE DESIGN DETAILS

A.1 INTRODUCTION

The PLL circuit is shown in detail in Figure 4. A listing of component values is found in Table III.

Two separate PLL circuits were designed and incorporated into the hardware; one for each of two frequency ranges. This was done in order to cover a very large total frequency range while maintaining fast response to changes in frequency [Ref. 7]. In the following sections the design procedure which was followed is documented.

A.2 DIGITAL PHASE LOCK LOOP (CMOS) DESIGN

The CD4046 digital PLL requires four areas of external design [Ref. 7].

(1) Selecting the timing capacitor C_1 which determines the center of the operating frequency range.

(2) Selecting the values of R_2 and ratio of R_1 to R_2 which determine the upper and lower bounds of the lock range.

(3) Selecting the ratios of R_3 to R_4, R_3 to C_2, and their values, which contribute to determine the damping ratio and settling time of the second order feedback loop.

(4) Interfacing the CMOS integrated circuit design with the TTL integrated circuits already in the PACER.

These areas are detailed in the following sections.

A.2.1 Timing Capacitor

In the following discussion, figures and pages are quoted with respect to Reference 8, the main source for design information. To begin the design a value of R2 was chosen within the limits listed on page 228 of Ref. 8. The value of C1 was approximated using figure 5(b) of Ref. 8. The value was then readjusted after testing to compensate for the effects of the following component values.

A.2.2 R1/R2

The chosen frequency range (fmax/fmin) was used to enter figure (c) of Ref. 8. The ratio R1/R2 was obtained from the data in that figure using the design value of the supply voltage to the PLL. Knowing the ratio R1/R2 and the value of R2 selected in section A.2.1, the value of R1 was obtained.

A.2.3 R3/R4/C2

The design of the loop low-pass filter was a trial and error iterative process because of effects from the counting circuits B1 and B2 present in the loop [Ref. 7]. The RC time constant of R3 and C2 determined the settling time of the loop while the ratio of R3 to R4 determined the damping ratio.

The nominal values found in Reference 7 were used initially and then these were adjusted to obtain what was considered to be the best loop response to changes in the input frequency. Loop response time was found by putting small but rapid perturbations on the test frequency, then noting the time to regain phase lock-on. By balancing the response time (required to be as fast as possible) against the settling time resulting from the loop damping ratio (at a minimum to maintain stability) across the frequency range, a satisfactory overall loop response was attained.

A.2.4 Interfacing

Due to the extremely high input and output impedances of CMOS integrated circuits, an interfacing buffer was needed between the CMOS PLL output from pin 3 and the TTL counter (B1) input to pin 5 (Fig. 2). Also, interface drivers were needed between the outputs of TLL counters B10 and B2 and the inputs to the CMOS PLL at pins 14 and 3, respectively.

The buffer between PLL pin 4 and counter B1 pin 5 simply required wiring one of the unused buffers which were part of the N4050B Hex buffer chip already in the PACER. Since the N4050B used a +5 VDC supply, the required transition from PLL +15V logic level to the counter +5v logic level was made.

In order to transition from the TTL (+5v) logic level of counters B10 and B2 outputs to the required PLL input levels (greater than +7v for logic 1), two 7417N TTL drivers

were used with 12 KΩ "pull up" resistors on their outputs. This gave a high logic level of +15v and a low state current drain on the drivers of only 1.25 ma each, well within their fan out capability [Ref. 9].

APPENDIX B
SOFTWARE DETAILS

This Appendix contains the following materials:

B.1 ACQUISITION FORTRAN PROGRAM A2D [Ref. 10]

B.1.1 Program A2D Flow Chart

B.1.2 Program A2D Listing

B.1.3 Program A2D Parameter Listing

B.2 SOFTWARE DRIVERS [Ref. 11, See Note 1]

B.2.1 Flow Chart

B.2.2 Pacer Driver DVR 70

B.2.3 A/D Driver DVR 56

<u>Notes on Software Drivers</u>

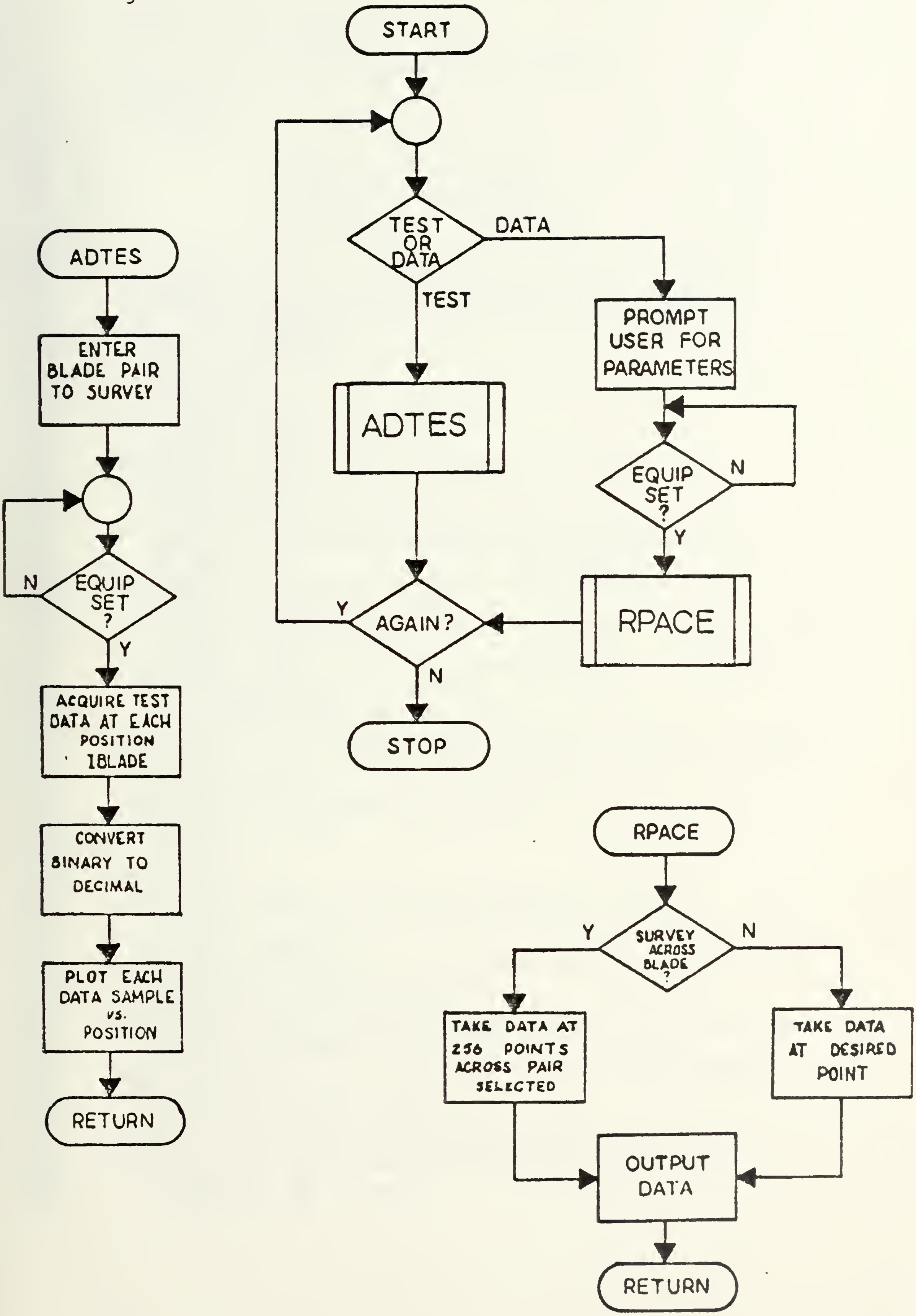
1. <u>Copyright</u>: The drivers DVR 70 and DVR 56 are copyrighted by the Hewlett-Packard Company, 1978. Approval for reproduction granted by Hewlett-Packard 22 May, 1981.
2. The driver flow chart in B.2.1 is a simplified diagram which shows the basic process for a typical driver. DVR 70 contains a series of steps which pass IBLADE (output) and a section which receives IRPM (inputs). The initiator section first outputs IBLADE to the PACER. After that, control is returned to the Central Interrupt Controller to await the PACER interrupt signal indicating it has IRPM ready to output. When the interrupt occurs, the completion section of DVR 70 is entered and IRPM is passed.

DVR 56, on the other hand, has only the input function to complete. It accomplishes this task as the standard driver indicated in the flow chart B.2.1. The beginning of DVR 56 configures the DMA feature of the RTE-IVB [Ref. 2].

B.1 ACQUISITION FORTRAN PROGRAM A2D

(See following pages).

B.1.1 Program A2D Flowchart



B.1.2 Program A2D Listing

```
AA2D   T=00004 IS ON CR00028 USING 00009 BLKS R=0000

FTN4,L
      PROGRAM A2D
C   .....................................................
C   .                                                   .
C   .                                                   .
C   .             PACED DATA ACQUISITION                .
C   .                                                   .
C   .           OPERATION AND TEST PROGRAM              .
C   .                                                   .
C   .          P. A. MCCARVILLE  APRIL 1981             .
C   .                                                   .
C   .....................................................
      COMMON IRPM
      INTEGER CHANL,AVERG,SURVEY,MODE,PAIR,POSIT,OFFSET
   90 WRITE (1,95)
   95 FORMAT(" WILL THIS BE SYSTEM TEST OR DATA RUN ?",
     *"  1=DATA 0=TEST")
      READ (1,*)ITEST
      IF (ITEST .EQ. 1) GO TO 96
      CALL ADTES(IGCB)
      CONTINUE
      GO TO 999
   96 WRITE (1,97)
   97 FORMAT("  ENTER TEST NUMBER ")
      READ (1,*)N2
   99 WRITE (1,100)
  100 FORMAT("  DO YOU WISH PROMPTING ?  1=YES 0=NO")
      READ (1,*)N1
      IF (N1 .EQ. 1) GO TO 102
  101 WRITE (1,110)
  110 FORMAT("  ENTER  CHANL,SURVEY,PAIR,POSIT,AVERG,OFFSET")
      READ (1,*) CHANL,SURVEY,PAIR,POSIT,AVERG,OFFSET
      GO TO 107
  102 WRITE (1,120)
  120 FORMAT("  ENTER DATA CHANNEL.  LIMITS 0-15")
      READ (1,*)CHANL
      WRITE (1,112)
  112 FORMAT("  ENTER # DATA SAMPLES TO AVERAGE.  LIMITS 1-99")
      READ (1,*) AVERG
      WRITE (1,130)
  130 FORMAT("  DO YOU WISH A SURVEY OR SINGLE PT ?  1=SURV",
     *"  0=SINGLE")
      READ (1,*) SURVEY
      IF (SURVEY .EQ. 1) GO TO 104
      WRITE (1,103)
  103 FORMAT("  WHICH BLADE PAIR DO YOU WISH TO SEE ?  LIMITS",
     *" 1-9")
      READ (1,*) PAIR
      WRITE (1,113)
  113 FORMAT("  WHICH POSITION BETWEEN BLADE PAIR ? LIMITS ",
     *"1-256")
      READ (1,*) POSIT
      MODE=1
      GO TO 107
  104 MODE=1
      WRITE (1,145)
  145 FORMAT("  WHICH BLADE PAIR DO YOU WISH TO SEE ?  LIMITS",
     *"  1-9")
      READ (1,*) PAIR
      WRITE (1,165)
  165 FORMAT("  DO YOU WANT TO OFFSET THE SURVEY ? 1=YES 0=NO")
      READ (1,*) OFFSET
      IF (OFFSET .EQ. 0) GO TO 107
      WRITE (1,166)
  166 FORMAT("  ENTER % OFFSET. (WILL DELAY START % OF 256)"
     *"  CHOICES- 50, 25, 12, OR 6")
      READ (1,*) OFFSET
  107 WRITE (1,170)
  170 FORMAT("  IS A/D CONVERTER ON ? IS TEST SET-UP READY ?",
     *"  1=YES  0=NO")
      READ (1,*)N8
      IF (N8 .EQ. 0) GO TO 107
  175 CALL RPACE (CHANL,AVERG,SURVEY,MODE,PAIR,POSIT,OFFSET,N2)
  176 WRITE (1,177)
  177 FORMAT("  DO YOU WISH ANOTHER RUN ?  1=YES 0=NO")
      READ (1,*) N3
      IF (N3 .EQ. 1) GO TO 90
  178 WRITE (6,168)
```

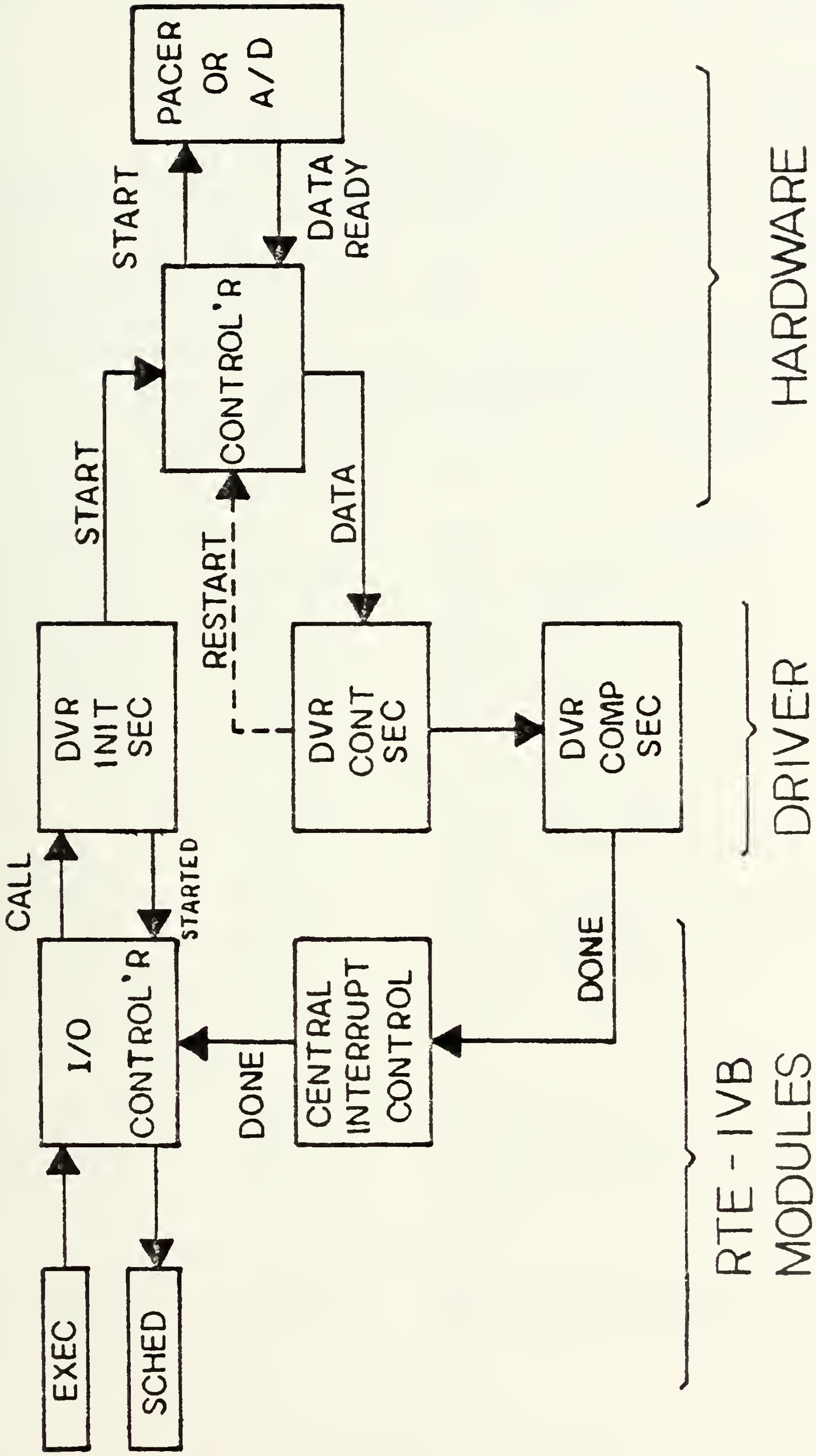
```
  160 FORMAT(25X,">>>  END OF RUNS  <<<",//)
  999 STOP
      END
      SUBROUTINE RPACE (ICHAN,IAVG,ISURV,IMODE,IPAIR,
     CIPOSIT,IOFFS,N2)
C     ....................................................
C     .                                                  :
C     .        DATA ACQISITION SUBROUTINE                :
C     .                                                  :
C     ....................................................
      REAL SRVPT(256)
      DIMENSION IBUFF(99),ITIME(5)
      N=IAVG
      CALL EXEC (11,ITIME,IYEAR)
      WRITE (6,90) N2,ITIME(5),IYEAR
   90 FORMAT(//,10X,"  THIS IS TEST #",I2," RUN ON JULIAN",
     *" DATE ",I3,3X,I4,//)
      IF (ISURV .EQ. 1) GO TO 120
C     ....................................................
C     .                                                  :
C     .          SINGLE POINT ACQUISITION                :
C     .                                                  :
C     ....................................................
      IBLADE=256*(IPAIR-1)+IPOSIT
      IF (IMODE .EQ. 0) GO TO 100
      IBLADE=IBLADE+100000B
  100 CALL EXEC (3,19)
      CALL EXEC (1,19,IRPM,1,IBLADE)
      CALL EXEC (1,20,IBUFF,N,ICHAN,0)
      DO 110 I=1,IAVG
  110 RBUFF=RBUFF+FLOAT(IBUFF(I))/32768.
      PTDATA=RBUFF/IAVG
      GO TO 195
C     ....................................................
C     .                                                  :
C     .     SURVEY ACROSS BLADE PAIR ACQUISITION         :
C     .                                                  :
C     ....................................................
  120 IF (IOFFS .EQ. 0) GO TO 125
      IOFFS=100/IOFFS
      GO TO 127
  125 IOFFS=1
  127 DO 140 J=1,256
      IBLADE=256*(IPAIR-1)+J+256/IOFFS+100000B
      CALL EXEC (3,19)
      CALL EXEC (1,19,IRPM,1,IBLADE)
      CALL EXEC (1,20,IBUFF,N,ICHAN,0)
      RBUFF=0.0
      DO 130 K=1,N
  130 RBUFF=RBUFF+FLOAT(IBUFF(I))/32768.
      DATA=RBUFF/IAVG
  140 SRVPT(J)=DATA
C     ..................................................
C     .                                                :
C     .             OUTPUT TABLES/PRINT                :
C     .                                                :
C     ..................................................
      WRITE (6,146)
  146 FORMAT(//,23X,"PACED SURVEY DATA",//)
      WRITE (6,148) SRVPT
  148 FORMAT(8(2X,F8.7))
  150 RPM=60/(IRPM*.000004)
      WRITE (6,165) RPM
  165 FORMAT(/,20X,"COMPRESSOR RPM FOR THIS RUN WAS ",
     CF7.2,/)
      GO TO 999
  195 WRITE (6,196) IPOSIT,IPAIR,PTDATA
  196 FORMAT(" THE DATA VALUE FOR POSITION ",I3," OF BLADE PAIR",
     *I2,"   IS   ",F10.7)
      GO TO 150
  999 RETURN
      END
      SUBROUTINE ADTES(IGCB)
*     .................................................:.
*     .                                                 :
*     .      TEST OF THE PACED DATA ACQUISITION         :
*     .                   SYSTEM                        :
*     ...................................................
      COMMON IRPM
      DIMENSION IGCB(192)
```

```
      INTEGER TIME(5),NOCR(2),IVOLT(128)
      REAL RVOLT(128)
      DATA NOCR /000033B,040433B/
      DATA ICHAN /0/
  101 FORMAT("  THIS WILL TEST THE PACED DATA SYSTEM FOR ",
     *"CONTINUITY AND LINEARITY.",/,"     ENTER THE SIMULATED",
     *" BLADE PAIR TO VIEW.  LIMITS 1-8")
      WRITE (1,101)
      READ  (1,  *) IPAIR
  100 WRITE (1,102)
  102 FORMAT("   IS THE TEST SET UP READY AS PER MANUAL ?",
     *"  1=YES  0=NO")
      READ (1,*) N4
      IF (N4 .EQ. 0) GO TO 100
      IBLADE = 0
      IBLADE = IBLADE+256*IPAIR
      IBLADE = IBLADE+100000B
      DO 128 I=1,128
      IBLADE=IBLADE+1
      CALL EXEC (3,19)
   20 CALL EXEC (1,19,IRPM,1,IBLADE)
  128 CALL EXEC(1,20,IVOLT(I),1,ICHAN,0)
      DO 45 J=1,128
   45 RVOLT(J)=FLOAT(IVOLT(J))/32768.
      LU=13
      ID=2
      CALL PLOTR(IGCB,ID,1,LU)
      CALL LIMIT(IGCB,0.,280.,0.,187.)
      CALL SETAR(IGCB,1.5)
      CALL VIEWP(IGCB,20.,140.,20.,80.)
      CALL WINDW(IGCB,0.,128.,0.,1.)
      CALL FXD(IGCB,1)
      CALL LGRID(IGCB,-2.,.05,0.,0.,8.,5.,1.)
      CALL MOVE(IGCB,1.,RVOLT(1))
      DO 55 K=2,128
      EX=FLOAT(K)
   55 CALL DRAW(IGCB,EX,RVOLT(K))
      CALL VIEWP(IGCB,0.,150.,0.,100.)
      CALL WINDW(IGCB,0.,150.,0.,100.)
      CALL MOVE(IGCB,64.,90.)
      CALL CPLOT(IGCB,-10,0.,0.)
      CALL LABEL(IGCB)
      WRITE(LU,160)
  160 FORMAT("PACED RAMP TEST DATA")
      CALL PLOTR(IGCB,ID,0)
      LU=6
      RETURN
      END
```

B.1.3 Program A2D Parameter Listing

CHANL/ICHAN	The A/D analog input channel to be sampled.
AVERG/IAVG	The number of samples per position to be averaged.
SURVEY/ISURV	Survey/single position selection
MODE/IMODE	Paced/free run-normally 1
PAIR/IPAIR	The pair of passages selected
POSI/IPOSIT	The position within the pair of passages
OFFSET/IOFFS	To start the survey later than position #1 within the pair passages. Entered as % of 256.
IRPM	See Table I
IBLADE	See Table I
IBUFF	The name of the set of digital data storage locations
N2	Test number that date
RBUFF	Floating point data storage
PTDATA/DATA	The data value at the selected point
SRVPT	The array holding the data surveyed
IGCB	Graphics control block, graphics package usage nonaccessible.

B.2 SOFTWARE DRIVERS



B.2.1 Flow Chart

B.2.2 Pacer Driver DVR 70


```
                    ASMB,R,L
      00000                    NAM DVR70,0 NPGS RPACE RTE DRIVER REV. 780724 JDM
                               ENT I.70,C.70,C.XX,I.XX
*
*     DRIVER FOR NAVAL POSTGRADUATE SCHOOL MOTEREY CA.
*     AUTHOR: JIM MORRIS  NEELY SANTA CLARA  408-996-9800
*
*     THIS RTE DRIVER WILL OUTPUT A BLADE NUMBER TO
*     THE PACER AND RETURN THE RPM VALUE.
*
*     CALLING SEQUENCE:
*     CALL EXEC(1,LU,IRPM,LEN,IBLADE)  >>>>> NORMAL INPUT (READ)
*               LU     = LOGICAL UNIT NUMBER OF PACER
*               IRPM   = RETURNED RPM VALUE FROM PACER
*               LEN    = 1 (NORMAL), = 0 (NOP - IMMED. COMPLETION)
*               IBLADE = PACER BLADE NUMBER (16 BIT INTEGER)
*
*     CALL EXEC(3,LU)  >>>>> CLEAR CONTROL ON PACER
*
      00000 000000  I.70  NOP              ENTER INITIATION SECTION
      00001 016064R       JSB SETIO        CONFIGURE I/O INSTRUCTIONS
      00002 161665        LDA EQT6,I       GET CONTROL WORD
      00003 012105R       AND B3           ISOLATE REQUEST TYPE
      00004 052106R       CPA B1           INPUT?
      00005 026023R       JMP D.X1          YES - DOIT
      00006 052105R       CPA B3           CONTROL?
      00007 026012R       JMP CNTRL         YES - DOIT
      00010 062106R       LDA B1           A=1 I.E. WRITE TO DEVICE NOT ILLEGAL
      00011 126000R       JMP I.70,I       ERROR RETURN TO IOC
*
* CHECK FOR CLEAR CONTROL AND NO SUBFUNTION BITS SET
*
      00012 161665  CNTRL LDA EQT6,I       GET CONTROL WORD
      00013 012110R       AND B3700        ISOLATE SUBFUNCTION BITS
      00014 002002        SZA              ANY SET?
      00015 026021R       JMP REJCT         YES, REJECT AS ILLEGAL CONTROL REQUEST
      00016 106700  I.0   CLC SC            NO, CLEAR DEVICE AND RETURN
      00017 062111R RTRN  LDA B4           IMMEDIATE COMPLETION A=4
      00020 002001        RSS               SKIP LOAD OF ERROR CODE A=2 (BAD CONTRO
      00021 062107R REJCT LDA B2           REJECT ERROR A=2
      00022 126000R       JMP I.70,I       RETURN TO IOC
*
* PROCESS READ REQUEST
*
      00023 161667  D.X1  LDA EQT8,I       GET BUFFER LENGTH
      00024 002002        SZA              CHECK IF = 0
      00025 026030R       JMP D.X3          NO, NORMAL PROCESS (1 WORD WILL BE INPU
      00026 006400        CLB                YES, B=0 (TRANSMISSION LOG)
      00027 026017R       JMP RTRN          RETURN TO IOC
*
* SETUP CONTINUATOR TO RETURN THIS SECTION
*
      00030 062104R D.X3  LDA P2           ADJUST ADDRESS
      00031 072035R       STA C.70         STUFF INTO CONTINUATOR RETURN
      00032 026046R       JMP D.X2         ENTER CONTINUATOR SECTION
```

```
*
* NORMAL RETURN TO IOC NOW
*
      00033 002400  IEXIT CLA             A=0 (ALL IS WELL)
      00034 126000R       JMP I.70,I      RETURN TO IOC
*
* CONTINUATION/COMPLETION SECTION
*
      00035 000000  C.70  NOP             ENTER CONT.
      00036 016064R       JSB SETIO       CONFIGURE I/O
      00037 161660        LDA EQT1,I      CHECK FOR SPURIOUS INTERRUPT
      00040 012112R       AND MASK        ISOLATE I/O REQUEST LIST POINTER
      00041 002002        SZA             IS A REQUEST IN PROGRESS?
      00042 026054R       JMP I.3          YES, GO DO IT
      00043 171774        STA EQT15,I      NO, ZERO TIME-OUT CLOCK
      00044 036035R       ISZ C.70        ADJUST RETURN TO P+2 (CONTINUATION)
      00045 126035R       JMP C.70,I      RETURN TO CIC
*
* OUTPUT CONTROLL WORD (BLADE NUMBER)
*
      00046 161670  D.Y2  LDA EQT9,I      GET BLADE NUMBER
      00047 000000  I.XX  NOP             <<<<< DEBUG ENTRY POINT >>>>>
      00050 102600  I.1   OTA SC          OUTPUT TO DEVICE
      00051 103700  I.2   STC SC,C        TURN ON DEVICE
      00052 036035R       ISZ C.70        ADJUST RETURN TO P+2 (CONTINUATION)
      00053 126035R       JMP C.70,I      RETURN TO CIC
*
* COMPLETION SECTION
*
      00054 102500  I.3   LIA SC          GET RPM FROM PACER
      00055 165656        LDB EQT7,I      GET RPM BUFFER ADDRESS
      00056 000000  C.YX  NOP             <<<<< DEBUG ENTRY POINT >>>>>
      00057 170001        STA B,I         STUFF INTO USER BUFFER
      00060 002400        CLA             SET A=0, ALL IS WELL RETURN CODE
      00061 006404        CLB,INB         SET B=1, TRANSMISSION LOG (1 WORD INPUT)
      00062 106700  I.4   CLC SC          CLEAR DEVICE
      00063 126035R       JMP C.70,I      RETURN TO CIC, COMPLETE
*
* CONFIGURE I/O INSTRUCTIONS
*
      00064 000000  SETIO NOP             ENTRY TO SUBROUTINE
      00065 052102R       CPA PIO         A=SC OF I/O DEVICE, CHECK IF CONFIGURED
      00066 126064R       JMP SETIO,I      YES, BYPASS CONFIGURATION
      00067 072102R       STA PIO         SAVE CURRENT I/O CHANNEL NUMBER
      00070 032103R       IOR LIA         COMBINE LIA WITH I/O
      00071 072054R       STA I.3         STORE IT
      00072 042113R       ADA B100        MAKE OTA INSTRUCTION
      00073 072050R       STA I.1         STORE IT
      00074 042114R       ADA B1100       MAKE STC,C INSTRUCTION
      00075 072051R       STA I.2         STORE IT
      00076 032115R       IOR B4000       MAKE CLC INSTRUCTION
      00077 072016R       STA I.0         STORE IT
      00100 072062R       STA I.4          AND AGAIN
      00101 126064R       JMP SETIO,I     RETURN FROM SUBROUTINE
*
```

```
* CONSTANTS/STORAGE/LINKS
*
00000              A      EQU 0
00001              B      EQU 1
00102 000000       PIO    OCT 0              CURRENT I/O SELECT CODE VALUE
00000              SC     EQU 0              DUMMY SELECT CODE
00103 102500       LIA    LIA SC          INPUT FROM DEVICE INSTRUCTION
00104 000032R      P2     DEF IEXIT-1     RETURN POINT IN INITIATION SECTION
00105 000003       B3     OCT 3
00106 000001       B1     OCT 1
00107 000002       B2     OCT 2
00110 003700       B3700  OCT 3700
00111 000004       B4     OCT 4
00112 077777       MASK   OCT 77777       MASK OFF BIT 15
00113 000100       B100   OCT 100
00114 001100       B1100  OCT 1100
00115 004000       B4000  OCT 4000
*
* BASE PAGE COMMUNITIONS AREA DEFINITIONS
*
01650              .      EQU 1650B       DEFINE START OF COMM AREA
01660              EQT1   EQU .+8
01661              EQT2   EQU .+9
01662              EQT3   EQU .+10
01663              EQT4   EQU .+11
01664              EQT5   EQU .+12
01665              EQT6   EQU .+13
01666              EQT7   EQU .+14
01667              EQT8   EQU .+15
01670              EQT9   EQU .+16
01671              EQT10  EQU .+17
01672              EQT11  EQU .+18
01771              EQT12  EQU .+81
01772              EQT13  EQU .+82
01773              EQT14  EQU .+83
01774              EQT15  EQU .+84
00116              SIZE   EQU *
                          END
**  NO ERRORS *TOTAL **RTE ASMB 92067-16011**
```

B.2.3 A/D Driver DVR 56

```
&DVR56 T=00003 IS ON CR00002 USING 00024 BLKS R=0000

ASMB,R,L,B,C       DVR56      JUNE,71
      HED <2310/2311 SUBSYSTEMS RTE DRIVER>
      NAM DVR56
      ENT I.56,C.56
      SPC 1
*  FORTRAN CALL:  CALL EXEC (1,IDRT,IBUFF,N,ICHAN,ICODE)
*    IDRT    SUBSYSTEM DEVICE REFERENCE NUMBER
*    IBUFF   INTEGER ARRAY (DATA STORAGE BUFFER)
*    N       NUMBER OF CONVERSIONS (DATA POINTS)
*    ICHAN   CHANNEL NUMBER
*    ICODE:  SUBSYSTEM/MODE:
*       0        2311 DIG ENCODE
*       1        2311 DIG PACE
*       2        2311 SEQ ENCODE
*       3        2311 SEQ PACE
*       4        2311 DIG FREE
*       5        2311 SEQ FREE
*       6        2310 DIG
*       7        2310 SEQ
      SPC 1
*  INITIATION SECTION
      SPC 1
I.56  NOP
      SPC 1
*  CONFIGURE INITIATION SECTION IO
      SPC 1
      STA B          SAVE IO ADDRESS
      IOR OTA        CONFIGURE
      STA IO12
      STA IO1         A/D
      STA IO13
      STA IO9          CONVERTER
      STA IO15
      STA IO16
      ADA =B300         IO
      STA IO7            INSTRUCTIONS
      ADA =B600
      STA IO10
      STA IO14
      LDA CHAN
      IOR OTA
      STA IO2        CONFIGURE
      ADA =B1100      DMA
      STA IO8           IO
      ADA =B176774       INSTRUCTIONS
      STA IO5
      ADA =B4000
      STA IO3
      XOR =B4100
      STA IO4
      STA IO6
      ADA =B4104
      STA IO11
      SKP
*  VALID REQUEST CHECK
      SPC 1
      LDA EQT6,I     READ
      CPA =B1
      JMP *+3        YES
ERROR CLA,INA        NO - REJECT
      JMP I.56,I     RETURN
      LDA EQT8,I     NUMBER OF REQUESTED
      CMA,INA         DATA POINTS GREATER
      SSA,RSS         THAN ZERO?
      JMP ERROR      NO - GO TO REJECT
      SPC 1
*  CONSTRUCT DMA CONTROL WORD
      SPC 1
      CLA,CCE        INITIALIZE SWITCH
      STA W1011       TO 2310 OPERATION
      LDA B          IO ADDRESS INTO A
      ADA =B20000    ADD CLC OPTION
      LDB EQT10,I    CODE WORD INTO B
      ADB =D-6       6 OR 7? I.E., 2310?
      CCE,SSB         YES
      JMP .2311       NO, 2311 OPERATION
      STB D0.S1      SET TO 2310 SEQ OR DIG MODE
```

```
        ELA,RAR       ADD STC OPTION
        JMP IO2
        SPC 1
.2311   STB W1011     SET SWITCH TO 2311 OPERATION
        LDB EQT10,I   CODE WORD INTO B
        CPB =B2       IF CODE 0 OR 2,
        RSS            ADD STC OPTION
        SZB,RSS
        ELA,RAR
        STA TEMP      SAVE DMA CONTROL WORD
        SPC 1
*  CONSTRUCT A/D CONTROL WORD
        SPC 1
        LDA EQT9,I    CHANNEL # TO A
IO16    OTA A.2.D
        CPB =B4       ICODE  COMMAND      PROGRAM
        JMP AHEAD       *      *            *
        RBR,SLB         0    DIG ENCODE   0000CH
        JMP *+4         1    DIG PACE     0100CH
        RBR,SLB         2    SEQ ENCODE   040000
        JMP *+3         3    SEQ PACE     050000
        JMP SINGL       4    DIG FREE     0200CH
        RBR             5    SEQ FREE     070000
        SPC 1
        CLA
IO1     OTA A.2.D     RESET A/D CONVERTER
        LDA =B40000
        SPC 1
SINGL   RBL
      . SSB
        ADA =B10000   PACER ENABLE BIT
        RBR,SLB
AHEAD   ADA =B20000   FREE RUN BIT
        STA B
        SKP
*  OUTPUT COMMAND WORDS TO DMA AND A/D
        SPC 1
        LDA TEMP
IO2     OTA DMA       CW1 TO DMA
IO3     CLC DMA
        LDA EQT7,I    BUFFER ADDRESS TO A
        IOR =B100000  DMA INPUT BIT
IO4     OTA DMA       CW2 TO DMA
IO5     STC DMA
        LDA EQT8,I    WORD COUNT (BUFF LENGTH)
        CMA,INA       NEGATIVE TO OUTPUT TO DMA
IO6     OTA DMA       CW3 TO DMA
        CLF 0         TURN OFF INTERRUPT
        LDA W1011     2310 OR 2311 OPERATION?
        CCE,SZA,RSS    2311
        JMP .2310      2310
        LDA B
IO7     CLF A.2.D
IO9     OTA A.2.D     ACTIVATE
IO10    STC A.2.D      A/D CONVERTER
IO8     STC DMA,C     ACTIVATE DMA
        CLA
        CPA DUMMY
        JMP I.56,I    RETURN
IO11    CLC DMA
        LDB INTBA
        LDA CHAN
        CPA =D7
        INB
        LDA B,I
        IOR =B100000
        STA B,I
        STF 0
        CLA           NORMAL
        JMP I.56,I    RETURN
        SPC 1
.2310   LDA EQT9,I    CHANNEL # TO A
IO12    OTA A.2.D     OUTPUT RANDOM MODE
        ELA,RAR       SET DIGITIZE MODE
IO13    OTA A.2.D     OUTPUT DIG MODE
IO14    STC A.2.D,C   ACTIVATE 2310
        LDB DO.S1     DIG OR SEQ?
        SZB,RSS
        JMP IO15       DIGITIZE
        CLE,INA        SEQUENTIAL
```

```
        ELA,RAR
        IOR =B40000    SET SEQ MODE
IO15    OTA A.2.D      OUTPUT NEXT MODE
        JMP IO8
        SKP
*  COMPLETION SECTION
        SPC 1
C.56    NOP
        IOR CLC        CONFIGURE
        STA *+1         CLC DMA INSTRUCTION
        CLC DMA
        LDA EQT4,I     A/D ADDRESS
        AND =B77        TO A
        IOR OTA        CONFIGURE OTA A.2.D
        STA *+2         INSTRUCTION
        CLA,CCE        TURN OFF
        OTA A.2.D       PACER
        LDB EQT8,I     TRANSMISSION LOG
        ELB,RBR         TO B
        JMP C.56,I     RETURN COMPLETION
        SKP
*  CONSTANTS
CLC     CLC IO
OTA     OTA IO
DO.S1   BSS 1
W1011   BSS 1
        SPC 3
IO      EQU 0
A.2.D   EQU 0
DMA     EQU 0
TEMP    EQU C.56
A       EQU 0
B       EQU 1
        SPC 3
*  SYSTEM BASE PAGE COMMUNICATION AREA
        SPC 1
.       EQU 1650B
        SPC 1
EQT4    EQU .+11
EQT6    EQU .+13
EQT7    EQU .+14
EQT8    EQU .+15
EQT9    EQU .+16
EQT10   EQU .+17
        SPC 1
CHAN    EQU .+19
INTBA   EQU .+4
DUMMY   EQU .+55
        SPC 3
        END
```

APPENDIX C

PACED DATA ACQUISITION USERS MANUAL

The two sections of this Appendix describe the use of program A2D for both (C.1) System Verification and (C.2) Test Data Acquisition.

C.1 SYSTEM VERIFICATION

In order to verify the complete paced data acquisition system (software and hardware), the following steps should be followed using the equipment shown in Fig. C.1.

C.1.1 <u>Procedure</u>

A WaveTek 142 signal generator or equivalent should be used to drive the test pulse feature of the PACER.

(1) Connect the "sync" output of the signal generator to the "sync" input on the PACER panel (Fig. 7).

(2) Connect the 50 Ω output of the WaveTek to the A/D analog channel to be tested (normally 0) and to the oscilloscope.

(3) Turn on the A/D converter.

(4) Set the WaveTek panel switches to produce a ramp voltage of 1 volt maximum peak amplitude from the 50 Ω output.

(5) On the PACER front panel connect the jack marked "BL" INPUT to the jack marked "BL" OUTPUT. Do

the same for the jacks marked "REV" INPUT and "REV" OUTPUT.

(6) Make sure "PACER ON" switches are in the "ON" position.

(7) Ensure that the Card #3 with the frequency range encompassing the blade passing frequency set on the WaveTek generator is installed in the PACER. If necessary remove the front panel air vent and replace Card #3 with the proper range card. Card #3 is shown in Figure 7.

(8) Turn on the PACER power switch and verify that the red pilot lamp is lit on the front panel.

(9) Log on the 21MX computer following the directions in the TPL Data Acquisition Manual.

(10) Once logged on, mount cartridge 28. Turn on the plotter and select the desired pen. Call up the Acquisition (Fortran) Program A2D with the command RP, A2D. Run the program with the command RU, A2D. The interactive program will prompt the user for responses. The responses are explained in the prompts which are given at the terminal. The prompts are as follows:

(a) System test or data run: enter Ø.

(b) Simulated blade pair to survey: enter any number 1-8.

(c) Is test set-up ready: if yes-enter 1, if no-enter Ø.

After prompt (c) is answered yes, and if the test is successful, the plotter will plot the same ramp signal that was set on the oscilloscope in C.1.1 step 4 (Fig. 12). The linearity and smoothness of the ramp signal indicate the degree to which data acquired under pacer control agree with the analog data input to the A/D converter.

C.2 TEST DATA ACQUISITION

In order to acquire paced data from the compressor (or other) test rig, the following steps should be followed with the equipment shown in Fig. C.1.

C.2.1 Procedure

(1) Cables to the PACER from the optical timing wheel on the test machine should be connected as shown in Fig. C.2. Verify the transducer input connections to the A/D converter at the A/D junction box.

(2) Turn on the A/D converter. Turn on the signal conditioner.

(3) Log on the 21MX computer following directions in the TPL Data Acquisition User's Manual. Call the Acquisition (Fortran) Program A2D by using the command RP, A2D. Then run the program by issuing the command RU, A2D.

(4) The interactive program will prompt the user for the following:

(a) System Test or Data Run: enter 1.

(b) Test number - enter integer.

(c) Do you wish prompting: Yes - enter 1, no - enter Ø.

From this point on, the program prompts are self-explanatory.

(5) At the completion of the data acquisition, the data values are printed out as shown in Table C-I.

(6) The final prompt will ask if another run is desired.

C.2.2 Data Storage

The survey data acquired in the program A2D is contained in the data memory locations SRUPT (J) where J = 1 - 256. The program A2D may be modified to output the data as desired by the user or to pass the data to a user-written subroutine for analysis.

Table C-1. Paced Data Output from Program A2D

THIS IS TEST # 5 RUN ON JULIAN DATE 149 1981

PACED SURVEY DATA

.0742188	.0937500	.1093750	.1210938	.1386719	.1582031	.1738281	.1894531
.1953125	.2148438	.2304688	.2500000	.2636719	.2734375	.2929688	.3046875
.3203125	.3359375	.3496094	.3652344	.3808594	.3925781	.4140625	.4277344
.4414063	.4570313	.4746094	.4882813	.5019531	.5097656	.5312500	.5488281
.5605469	.5742188	.5917969	.6074219	.6171875	.6386719	.6523438	.6660156
.6835938	.6972656	.7050781	.7207031	.7402344	.7558594	.7734375	.7832031
.7988281	.8164063	.8320313	.8515625	.8593750	.8828125	.8964844	.9042969
.9179688	.9335938	.9492188	.9667969	.9765625	.9980469	.9824219	.9746094
.9570313	.9375000	.9218750	.9082031	.8925781	.8769531	.8613281	.8535156
.8320313	.8144531	.8046875	.7851563	.7753906	.7558594	.7402344	.7324219
.7109375	.6972656	.6796875	.6699219	.6523438	.6347656	.6250000	.6074219
.5937500	.5742188	.5585938	.5468750	.5292969	.5175781	.5078125	.4921875
.4667969	.4609375	.4453125	.4238281	.4062500	.3945313	.3808594	.3691406
.3535156	.3417969	.3222656	.3066406	.2968750	.2753906	.2636719	.2460938
.2363281	.2187500	.1992188	.1875000	.1699219	.1582031	.1445313	.1269531
.1074219	.1015625	.0859375	.0703125	.0527344	.0410156	.0488281	.0664063
.0761719	.0937500	.1132813	.1210938	.1347656	.1542969	.1679688	.1855469
.1992188	.2128906	.2285156	.2480469	.2636719	.2753906	.2890625	.3027344
.3203125	.3359375	.3457031	.3593750	.3769531	.3945313	.4160156	.4238281
.4375000	.4531250	.4726563	.4804688	.5019531	.5097656	.5351563	.5410156
.5585938	.5781250	.5820313	.6093750	.6191406	.6367188	.6523438	.6582031
.6738281	.6933594	.7109375	.7226563	.7402344	.7597656	.7675781	.7871094
.7968750	.8144531	.8281250	.8457031	.8574219	.8750000	.8925781	.9042969
.9218750	.9414063	.9570313	.9687500	.9804688	.9960938	.9902344	.9765625
.9492188	.9355469	.9218750	.9082031	.8925781	.8710938	.8613281	.8437500
.8281250	.8203125	.8046875	.7871094	.7695313	.7617188	.7460938	.7265625
.7187500	.6972656	.6835938	.6699219	.6562500	.6406250	.6230469	.6132813
.5937500	.5761719	.5644531	.5429688	.5273438	.5195313	.5097656	.4960938
.4765625	.4550781	.4414063	.4257813	.4179688	.3945313	.3847656	.3652344
.3554688	.3398438	.3222656	.3085938	.2968750	.2792969	.2675781	.2441406
.2363281	.2187500	.2031250	.1855469	.1699219	.1542969	.1425781	.1269531
.1093750	.0996094	.0859375	.0703125	.0566406	.0429688	.0449219	.0644531

COMPRESSOR RPM FOR THIS RUN WAS 18541.4

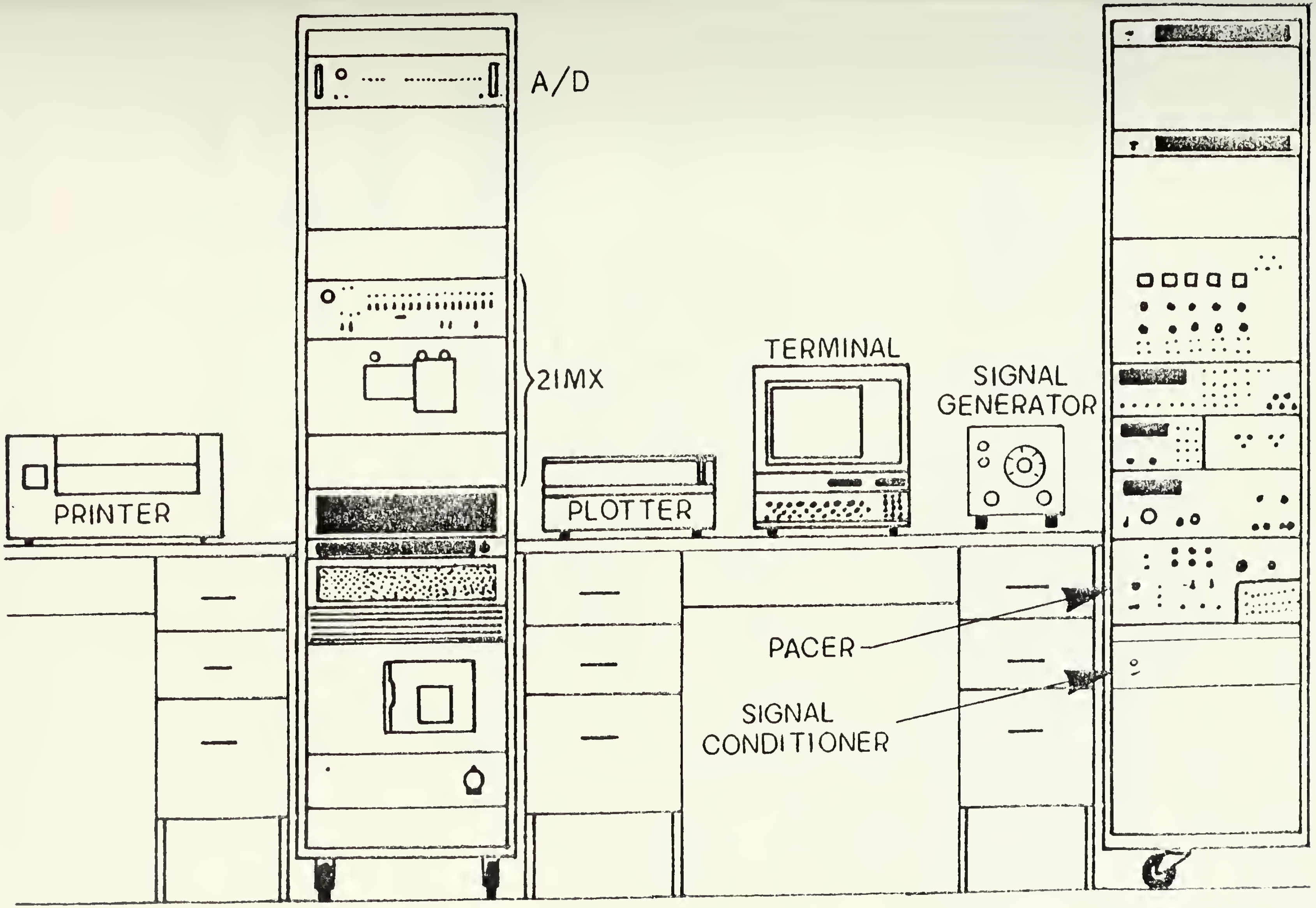


Figure C1. Data Acquisition and Test Equipment

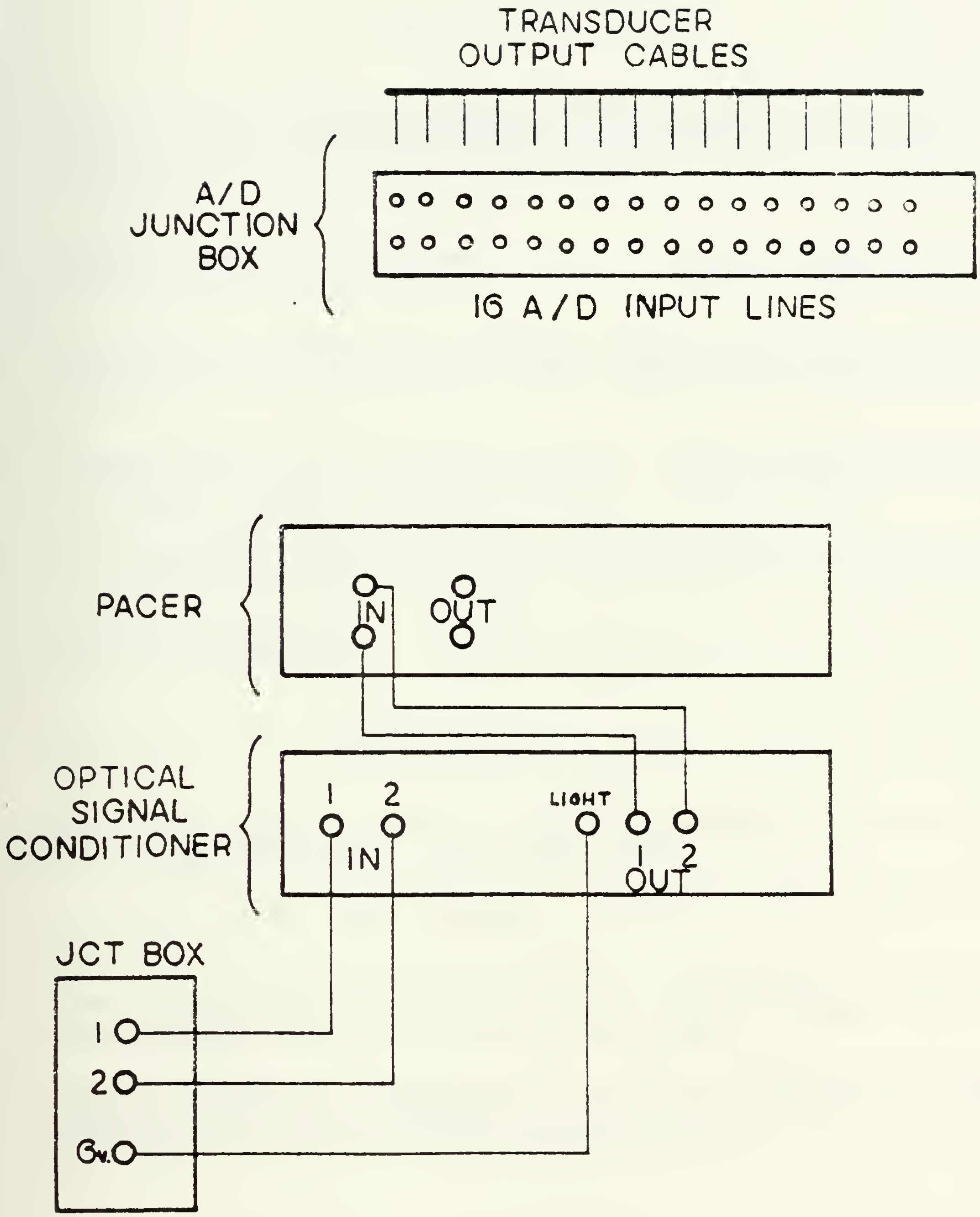


Figure C2. Cable Connections for Test Data Acquisition

LIST OF REFERENCES

1. West, J. C., Digital Programmable Timing Device for Fast Response Instrumentation in Rotating Machines, M.S.A.E. Thesis, Naval Postgraduate School, Monterey, California, 1976.

2. Hewlett-Packard Reference Manual, HP21MX Computer Series, HP02108-90002, June 1976.

3. Geopfarth, R. N. Introductory Guide for Users of RTE-IV, TPL Technical Note 79-01, TPL, Monterey, California, 1979.

4. Hewlett-Packard Reference Manual, RTE-IVB Programmer's Reference Manual, HP92068-90004, January 1980.

5. Hewlett-Packard Operating and Service Manual, Analog to Digital Converter, 5610A, 05610-91999, January 1972.

6. Signetics Catalog, Signetics Corporation, Copyright 1977.

7. Lancaster, D., CMOS Cookbook, Howard W. Sams & Co. Inc., 1977.

8. RCA Solid State Databook, COS/MOS Digital Integrated Circuits, SSD-203B, Copyright 1973.

9. Lancaster, D., TTL Cookbook, Howard W. Sams & Co. Inc., 1974.

10. Hewlett-Packard Reference Manual, RTE FORTRAN IV Reference Manual, HP92060-90023, March 1980.

11. Hewlett-Packard Programming and Operating Manual, RTE Operating System Drivers Writing Manual, HP92200-93005, May 1978.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93940	2
3.	Department Chairman, Code 67 Department of Aeronautics Naval Postgraduate School Monterey, CA 93940	1
4.	Director, Turbopropulsion Laboratory, Code 67Sf Naval Postgraduate School Monterey, CA 93940	1
5.	LCDR J. C. West 6229 Hanley Corpus Christi, TX 78412	1
6.	Turbopropulsion Laboratory Code 67 Naval Postgraduate School Monterey, CA 93940	15
7.	Dr. Gerhard Heiche Naval Air Systems Command Code AIR-310 Department of the Navy Washington, D.C. 20360	1
8.	Dr. A. D. Wood Office of Naval Research Eastern/Central Regional Office 666 Summer Street Boston, MA 02210	1
9.	LCDR P. A. McCarville 8151 Alton Dr. Lemon Grove, CA 92045	1

Thesis 193686
M1666 McCarville
c.1 Hardware and software improvements to a paced data acquisition system for turbomachines.